

**RECEP TAYYIP ERDOGAN UNIVERSITY**

**FACULTY OF ENGINEERING AND ARCHITECTURE**

**COMPUTER ENGINEERING DEPARTMENT**

**Data Mining**

**2024-2025**

**Student Name Surname**

Rıdvan Karasubaşı

**Student No**

201401030

**Topic**

Using and Analyzing Data Mining Methods

**Data Sets Used**

Heart Disease Dataset

Wholesale Customer Data

California Housing Database

**Applied Methods:**

Decision Trees and Random Forest Classification

K-Means Clustering

Linear Regression and Random Forest Regression

**Instructor**

Doctor Lecturer Abdulgani Kahraman

**RİZE**

**2024**

**1. Introduction**

This report is about using and analyzing basic methods in data mining. Data mining means finding important information from large and complex sets of data. It is very important in today’s world. Because there is a lot of data. This report looks at three main data mining methods and checks how they work on different data sets.

First, decision trees and random forest classification methods are used to predict a target variable that has categories. In this part data was cleaned, the model was tested, and its performance was compared. In the second part cluster analysis was done using the K-Means method to find natural groups in the data. Lastly, linear regression and random forest regression methods were compared to predict a continuous target variable.

The types of data sets used, the details of the methods, the ways to measure results, and the outcomes are all discussed in the report. The results are shown with visual tools and compared with each other. This report aims to help understand data mining techniques in theory and practice.

**2. Datasets Used**

This report uses three different datasets. Each dataset is analyzed with different data mining methods. The datasets and their features are shown below:

**2.1. Heart Disease Dataset**

**Source:** <https://archive.ics.uci.edu/dataset/45/heart+disease>

**Description:** This dataset has health informations from people with and without heart disease. It includes details like age, type of chest pain, and cholesterol levels.

**Why I Chose This Data Set:** This dataset does not contain any missing values and this data set is easy to understand. We did not eliminate any feature but we updated the target column.

**Target Variable(Outcome):**

* Target:
  + 1: Heart disease
  + 0: No heart disease

**Purpose:** To predict the target varible using decision tree and other classification algorithms.

**Details:**

* **Total Observations:** 303
* **Number of Columns:** 13
* **Independent Variables:**
  + **age:** Age of the individual.
  + **sex:** Gender(1: Male, 0: Female).
  + **cp:** Type of chest pain experienced(typical angina, atypical angina, non anginal pain, asymptomatic).
  + **trestbps:** Resting blood pressure.
  + **chol:** Serum cholesterol level.
  + **fbs:** Fasting blood. If sugar is bigger than 120 than 1(Yes), else 0(No).
  + **restecg:** Resting electrocardiographic results.
  + **thalach:** Maximum heart rate achieved.
  + **exang:** Exercise-induced angina(1: Yes, 0: No).
  + **oldpeak:** ST depression induced by exercise relative to rest.
  + **slope:** Slope of the ST segment during exercise.
  + **ca:** Number of major vessels visible via fluoroscopy.
  + **thal:** Thalassemia type(3: Normal, 6: Fixed defect, 7: Reversible defect).

**2.2. Wholesale Customer Data**

**Source:** <https://archive.ics.uci.edu/ml/datasets/Wholesale+customers>

**Description:** This dataset provides segmentation data for wholesale customers, including annual spending values on categories such as milk, grocery, and frozen products.

**Why I Chose This Data Set:** This dataset does not contain any missing values and this data set is easy to understand. We eliminated channel and region columns because they are not necessary for clustering.

**Target Variable:** None(Clustering analysis does not include a target variable.)

**Purpose:** To perform customer segmentation using the K-Means algorithm.

**Details:**

* **Total Observations:** 440
* **Number of Columns:** 8
* **Independent Variables:**
  + **Fresh:** Annual spending on fresh products.
  + **Milk:** Annual spending on milk products.
  + **Grocery:** Annual spending on grocery products.
  + **Frozen:** Annual spending on frozen products.
  + **Detergents\_Paper:** Annual spending on detergents and paper products.
  + **Delicassen:** Annual spending on delicatessen products.
  + **Region:** Geographic region (1: Lisbon, 2: Porto, 3: Other).
  + **Channel:** Type of channel (1: Hotel/Restaurant/Cafe, 2: Retail).

**2.3. California Housing Database**

**Source:** The dataset can be found in the scikit-learn library.

**Description:** This dataset contains features related to housing in California, such as income, number of rooms, and house age.

**Why I Chose This Data Set:** This dataset does not contain any missing values and this data set is easy to understand. We did not eliminate any feature.

**Target Variable:**

* **Price:** Median house value for California districts.

**Purpose:** To predict house prices using linear regression and advanced regression techniques.

**Details:**

* **Total Observations:** 20,640
* **Number of Columns:** 9
* **Independent Variables:**
  + **MedInc:** Median income in the block group.
  + **HouseAge:** Median house age in the block group.
  + **AveRooms:** Average number of rooms per household.
  + **AveBedrms:** Average number of bedrooms per household.
  + **Population:** Population in the block group.
  + **AveOccup:** Average number of occupants per household.
  + **Latitude:** Block group latitude.
  + **Longitude:** Block group longitude.

**3. Question 1: Decision Tree and Comparative Analysis**

**3.1. Importing Necessary Libraries**

The necessary libraries which I imported for this task are shown below:

**metin, yazı tipi, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu**

* **sklearn.model\_selection.train\_test\_split:** This library used for splitting data to training and test data.
* **sklearn.tree.DecisionTreeClassifier:** Decision tree classifier algorithm.
* **sklearn.tree.plot\_tree:** It visualize decision trees and it makes it easier to understand.
* **sklearn.ensemble.RandomForestClassifier:** Random forest classifier algorithm.
* **sklearn.metrics:** This library includes model evaluation metrics. The metrics are shown below:
  + **accuracy\_score:** It measures the accuracy rate of the model.
  + **precision\_score:** It measures the accuracy of the positive predictions.
  + **recall\_score:** It measures the rate of the true positive.
  + **f1\_score:** It calculates the harmonic mean of precision and recall.
  + **confusion\_matrix:** It summarizes the difference between actual values and predicted values of the model.
* **seaborn:** This library used for data visualization.
* **matplotlib.pyplot:** This library offers charting tools to visualize the data.
* **pandas:** This library used for data processing and data analyse.

**3.2. Data Preprocessing**

**3.2.1. Loading and Preparing the Data**

I loaded my dataset named heart.csv and then assigned the data to the heart\_data variable. Now heart\_data is a dataframe and it includes the all data in the csv file. Then, I create a map for simplify the target value. 1’s will be 1 and 2’s will be 0.

metin, yazı tipi, ekran görüntüsü, beyaz içeren bir resim

Açıklama otomatik olarak oluşturuldu

**3.2.2. Handling missing Values**

The dataset does not have any missing values. This makes it easier to use for training and testing the models.

**3.2.3. Feature Selection and Engineering**

* The target column is the dependent variable. It has two classes:
  + 1: Patient has heart disease.
  + 0: Patient does not have heart disease.
* All other columns are independent variables used to train the models.
* No unnecessary features were removed. All features were included in the analysis.

metin, yazı tipi, beyaz, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu

**3.3. Modeling**

**3.3.1. Splitting Data**

* I divided the data two times. In the first I select 80% training data for the model to learn and 20% test data for evaluating the model performance. In the second I select 70% training data for the model to learn and 30% test data for evaluating the model’s performance.
* The splitting was done using the train\_test\_split function with random\_state=58 to ensure reproducibility.

**First selection:**



**Second selection:**



**3.3.2. Decision Tree Classifier**

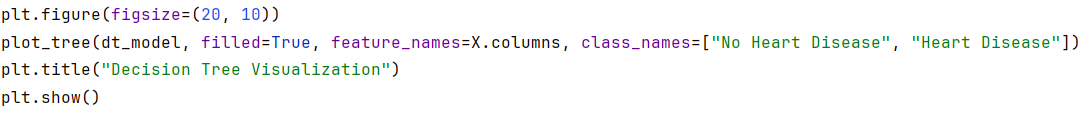
* A decision tree was used to predict the target variable.
* The DecisionTreeClassifier from Python’s sklearn library was applied to the dataset.
* The model was trained on the training dataset.

metin, ekran görüntüsü, yazı tipi içeren bir resim

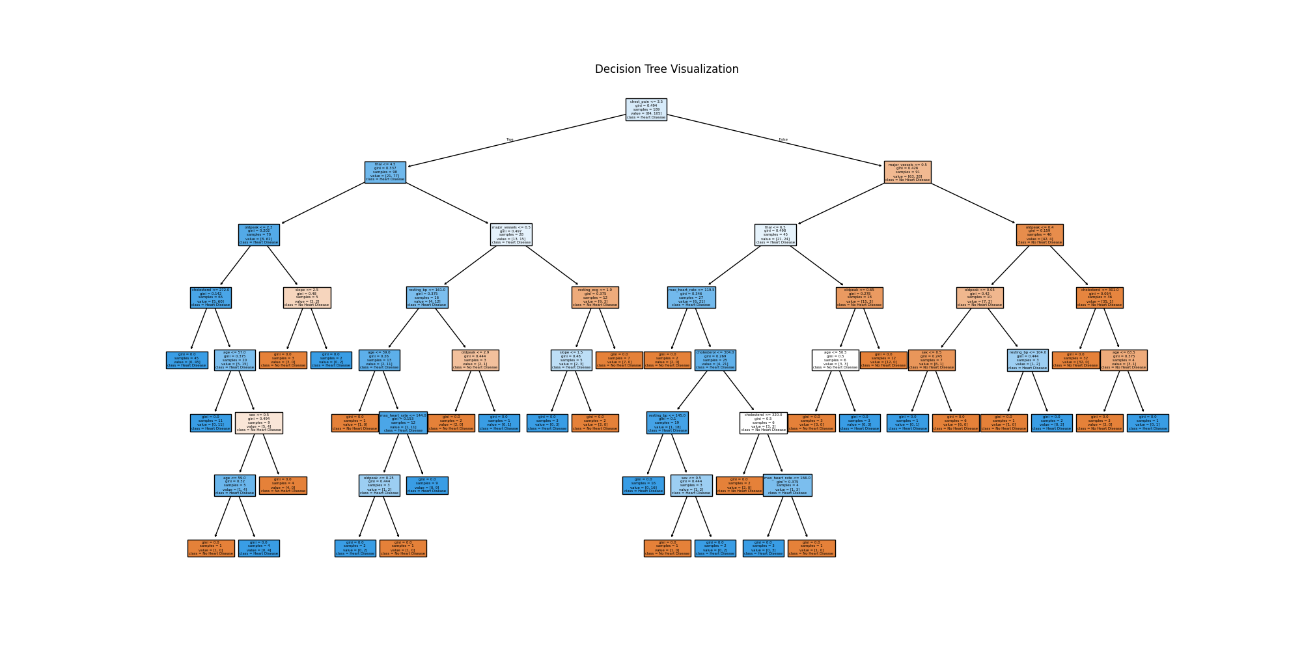
Açıklama otomatik olarak oluşturuldu

**Decision Tree Visualization:**

In the code below, we visualized the Decision Tree. We setted the figure size with using plt.figure function and then we visualized the decision tree with using plot\_tree function. We sent some parameters like feature\_names and class\_names. Lastly, we setted a plot title.



**Result:**



**3.3.3 Random Forest Classifier**

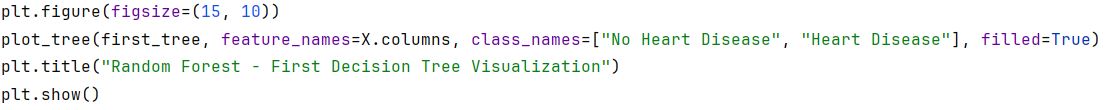
* A random forest was used to predict the target variable.
* The RandomForestClassifier from Python’s sklearn library was applied to the dataset.
* The model was trained on the training dataset.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Random Forest Visualization:**

In the code below, we visualized the Random Forest’s first decision tree. In total, 100 random forest decision trees were created. It was not possible to show all of them so I show only the first tree. We setted the figure size with using plt.figure function and then we visualized the random forest’s first decision tree with using plot\_tree function. We sent some parameters like feature\_names and class\_names. Lastly, we setted a plot title.



**Result:**

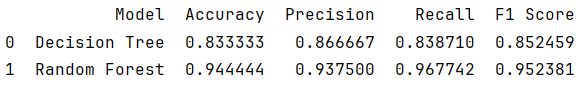
**diyagram, çizgi, plan, harita içeren bir resim

Açıklama otomatik olarak oluşturuldu**

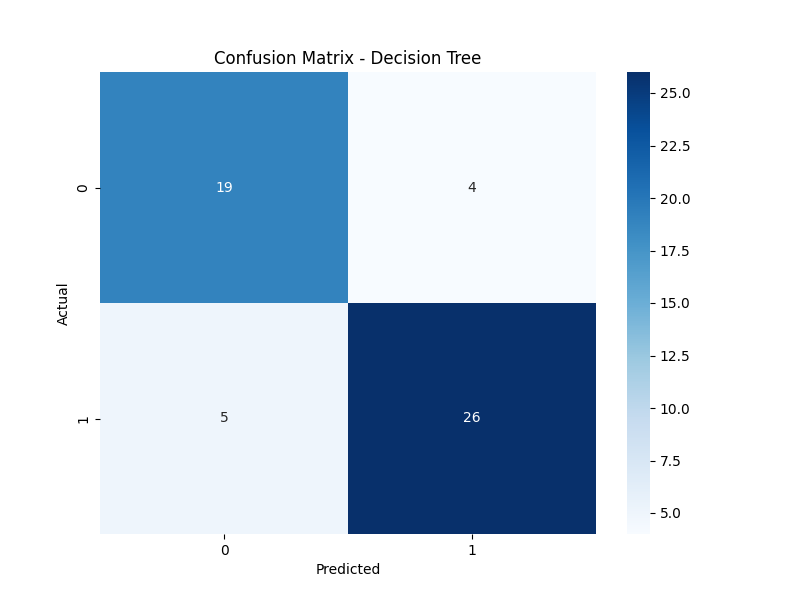
**3.4. Performance Evaluation**

**3.4.1. Decision Tree Performance**

The performance of two decision tree models was evaluated using accuracy, precision, recall, and F1 score for two selection. For first selection results are shown below:

****

**Confusion Matrix for Decision Tree:**



* **True Positives (Correctly predicted heart disease):** 26
* **True Negatives (Correctly predicted no heart disease):** 19
* **False Positives (Wrongly predicted heart disease):** 4
* **False Negatives (Wrongly predicted no heart disease):** 5

For second selection results are shown below:

**metin, yazı tipi, ekran görüntüsü, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**Confusion Matrix for Decision Tree:**

metin, ekran görüntüsü, dikdörtgen, diyagram içeren bir resim

Açıklama otomatik olarak oluşturuldu

* **True Positives:** 39
* **True Negatives:** 30
* **False Positives:** 6
* **False Negatives:** 6

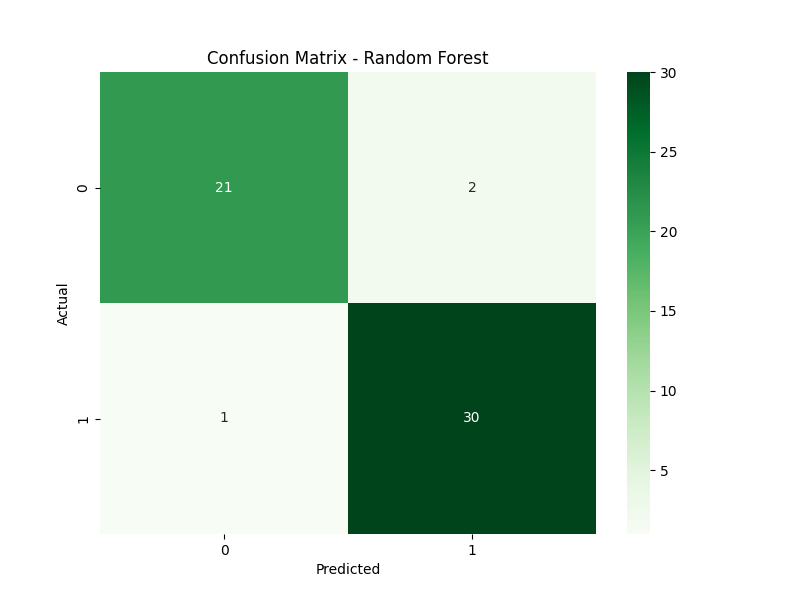
**3.4.2. Random Forest Performance**

A random forest classifier was used to compare with the decision tree model. For the first selection performance metrics are:

**metin, yazı tipi, ekran görüntüsü, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**Confusion Matrix for Random Forest:**



* **True Positives:** 30
* **True Negatives:** 21
* **False Positives:** 2
* **False Negatives:** 1

For the second selection performance metrics are:

**metin, yazı tipi, ekran görüntüsü, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**Confusion Matrix for Random Forest:**

ekran görüntüsü, dikdörtgen, yeşil, metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

* **True Positives:** 44
* **True Negatives:** 32
* **False Positives:** 4
* **False Negatives:** 1

**3.5. Results and Discussion**

**3.5.1. Comparison of Model Performance**

I evaluated the performance of decision tree classifier and random forest classifier with all metrics.

**Accuracy:** The random forest classifier achieved a higher accuracy rate than the decision tree model in both data separations. This shows that random Forest provides better generalization and reduces overfitting.

**Precision:** Random forest obtains higher precision value in both data separates. It means random forest is more effective to reducing the false positive than the decision tree.

**Recall:** True positive rate of the random forest is better than the decision tree's true positive rate.

**F1 Score:** The random forest classifier achieved a higher F1 score than the decision tree model in both data separations. F1 score is a very critical metric to evaluate the overall model’s performance.

**Confusion Matrix:** In random forest, false positive rate and false negative rate is less than the decision tree. This increased the overall accuracy and reliability of the model.

**Overall Evaluating:**

In general, random forest works better than decision tree. It is found that random forest can give more stable and reliable results because it uses more than one decision tree. So, the random forest might be a better choice for important classification problems like predicting heart disease.

**3.5.2. Why Random Forest Performs Better**

* **Ensemble Learning:** Random forest uses many decision trees to make predictions, which improves accuracy.
* **Lower Overfitting:** Random forest reduces overfitting compared to a single decision tree.
* **Better Generalization:** Random forest makes the model more strong and works better for test data.

**4. Question 2: K-Means Clustering**

**4.1 Importing Necessary Libraries**

The necessary libraries which I imported for this task are shown below:

**metin, ekran görüntüsü, yazı tipi, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

* **pandas:** This library used for data processing and data analyse.
* **sklearn.cluster.KMeans:** This library provides the K-Means algorithm.
* **sklearn.preprocessing.StandardScaler:** This library used for scaling the datas.
* **sklearn.metrics.silhouette\_score:** This library used for evaluation the clustering algorithms.
* **matplotlib.pyplot:** This library offers charting tools to visualize data.
* **seaborn:** This library used for data visualization.

**4.2. Data Preprocessing**

**4.2.1 Loading and Preparing the Data**

I loaded my dataset named wholesale\_customers\_data.csv. I dropped the Channel and Regions colums because they are not necessary for clustering.

**metin, yazı tipi, ekran görüntüsü, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**4.2.2. Scaling and Missing Values**

* The wholesale customers dataset was used for clustering.
* The Channel and Region columns were removed because they are not need for clustering.
* The dataset has no missing values.
* The features were scaled using StandardScaler. This helps the clustering algorithm by putting all features on the same scale. I create a scaler for scaling the features.
* It scales the data using fit\_transform function.

metin, yazı tipi, beyaz, tipografi içeren bir resim

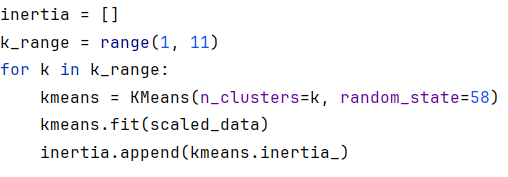
Açıklama otomatik olarak oluşturuldu

**4.3. Clustering Process**

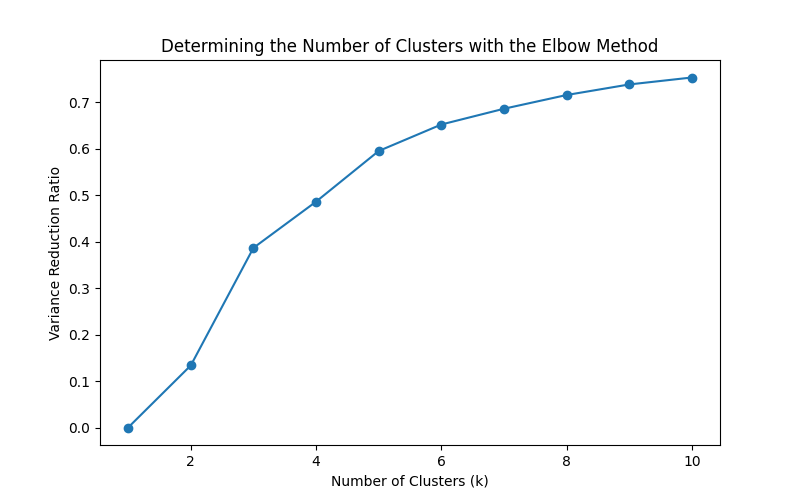
**4.3.1. Choosing the Number of Clusters(k)**

* The elbow method was used to find the best number of clusters.
* The elbow graph shows how much the variance changes for different k values(1 to 10).
* The “elbow point” shows the best number of clusters. The result is k = 3. As you see in the elbow graph, there is a big break at the point where k is 3. So, the optimal k value is 3. We can select the k as 3.

I created a empty list named internia. This list is used to store the internia values (total error sum of squares) obtained as a result of each clustering. It runs the K-Means algorithm for different cluster numbers and calculates the inertia values. Then, it assigns this value to the internia list. We will use these values ​​later when drawing the elbow graph.

****

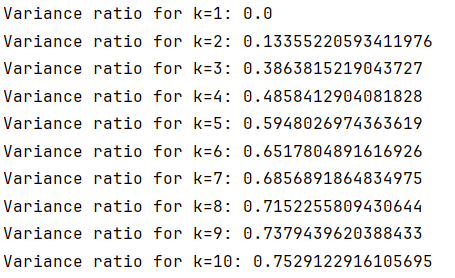
**Elbow Method Graph:**



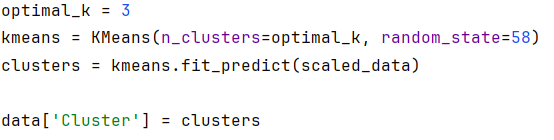
This code calculates the variance ratio using inertia values. The variance ratio indicates how well each clustering solution explains the data.



You can see the variance values in the graph which shown below. Variance ratio for k=2 is 0.13 and variance ratio for k=3 is 0.38. Almost 3 times. So we can say k=3 is elbow.



The optimal number of clusters determined as a result of the previous analysis was calculated as 3. A model created for K-Means algorithm and then K-Means algorithm runned and it separates data to the clusters. Finally, it creates a column named Cluster and it adds cluster values to the Clusters column.



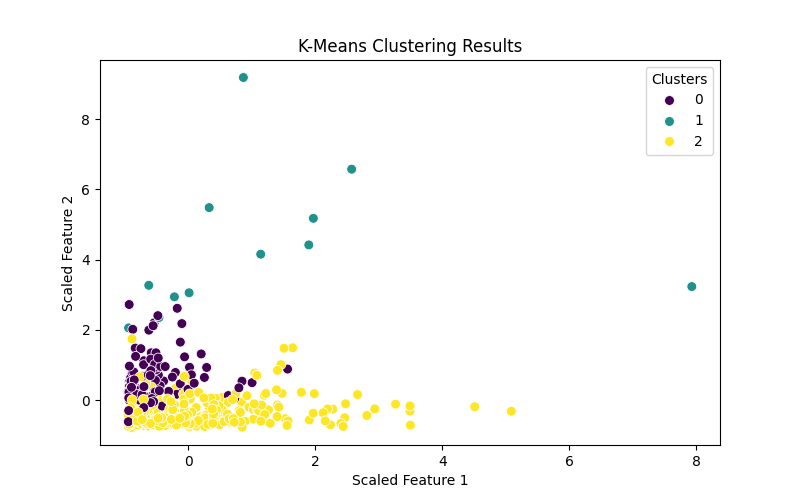
It calculates the silhouette score. Silhouette score will show how well separate the clusters.



**4.3.2. Cluster Results**

* The K-Means algorithm was used with k = 3 clusters.
* A scatter plot shows the clusters.
* Each dot is a customer, and each color is a different cluster.

**Cluster Results Scatter Plot:**



**4.4. Performance Evaluation**

* **Silhouette Score:** The silhouette score checks how good the clusters are. A higher score means better clustering.



**4.5. Results and Discussion**

**4.5.1. Understanding the Clusters**

* Cluster 0: Customer who spend a lot on categories like fresh or frozen products.
* Cluster 1: Customers who spend a medium amount on all categories.
* Cluster 2: Customers who spend less in most categories.

**4.5.2. Analysis**

* The K-Means algorithm worked well with this dataset.
* The elbow method and silhouette score show that k = 3 is the best choice.
* Businesses can user these clusters to understand customer spending behavior and improve their services.

***Conclusion***

K-Means clustering method was used to group customers into three main clusters based on their spending habits. The number of clusters was determined using the elbow method, which showed that three clusters would give the best results. Each cluster represents a different type of customer: high spenders, medium spenders, and low spenders. These groups help businesses to better understand their customers and make more informed decisions.

The elbow method helped identify the most suitable number of clusters by showing a significant drop in variance when the cluster number reached three. After determining this, the K-Means algorithm was applied, and the data was divided into these groups. Each cluster was visualized with a scatter plot, making it easy to see how customers were grouped based on their spending.

The analysis also included a silhouette score, which showed that the clustering worked well. The results are important for businesses because they can use this information to offer better services, target specific groups of customers, and create strategies for each cluster.

**5. Question 3: Regression Analysis**

**5.1. Importing Necessary Libraries**

The necessary libraries which I imported for this task are shown below:

**metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

* **pandas:** This library used for data processing and data analyse.
* **sklearn.datasets.fetch\_california\_housing:** This library loads the California Housing dataset.
* **sklearn.model\_selection.train\_test\_split:** This library used for splitting data to training and test data.
* **sklearn.ensemble.RandomForestRegressor:** This library solves the regression problem using the random forest algorithm.
* **sklearn.linear\_model.LinearRegression:** This library provides the linear regression model.
* **sklearn.metrics:** It provides some metrics for measuring of the model’s performance. The metrics are shown below:
  + **mean\_squared\_error(MSE):** Calculates the mean squared error. Smaller values ​​indicate a better model.
  + **mean\_absolute\_error(MAE):** Calculates the average of the absolute difference between predicted and actual values.
  + **r2\_score:** It measures the explanatory power of the model. 1 indicates the ideal model, 0 indicates the average model**.**
* **sklearn.preprocessing.StandardScaler:** This library used for scaling the datas.
* **matplotlib.pyplot:** This library offers charting tools to visualize data.
* **seaborn:** This library used for data visualization.

**5.2. Data Preprocessing**

**5.2.1. Loading Dataset and Preparing the Target Value**

* The California housing dataset was used.
* Dataset loaded from sklearn datasets library.
* I create x and y variables. X will store our data as a dataframe and y variable, will store the target variable. The target variable is Price, which represents the median house price in California.

**metin, yazı tipi, ekran görüntüsü, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**5.2.2. Feature Scaling and Missing Values**

* All features were scaled using StandardScaler to make their values comparable.
* There were no missing values in the dataset.
* The features were scaled using StandardScaler. This helps the clustering algorithm by putting all features on the same scale. I create a scaler for scaling the features.
* It scales the x using fit\_transform function.

metin, yazı tipi, beyaz, tipografi içeren bir resim

Açıklama otomatik olarak oluşturuldu

**5.3. Modeling**

**5.3.1. Splitting Data**

* I divided the data two times. In the first I select 80% training data for the model to learn and 20% test data for evaluating the model performance. In the second I select 70% training data for the model to learn and 30% test data for evaluating the model’s performance.
* The splitting was done using the train\_test\_split function with random\_state=58 to ensure reproducibility.

**First selection:**



**Second selection:**



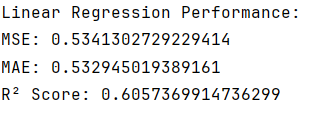
**5.3.2. Lineer Regression**

* A lineer regression model was trained on the dataset.
* It uses a straight line to predict house prices based on input features.

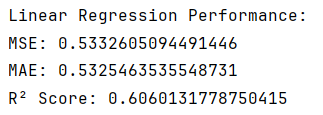
metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

The lineer regression’s performance for first selection is shown below:



The lineer regression’s performance for second selection is shown below:



**5.3.3. Random Forest Regression**

* A random forest regression model was used for comparison.
* Random forest is an advanced technique that uses multiple decision trees to make predictions.

metin, ekran görüntüsü, yazı tipi içeren bir resim

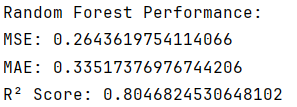
Açıklama otomatik olarak oluşturuldu

The random forest’s performance for first selection is shown below:

metin, yazı tipi, beyaz içeren bir resim

Açıklama otomatik olarak oluşturuldu

The random forest’s performance for second selection is shown below:



**5.4. Performance Evaluation**

The models were evaluated using three metrics:

* Mean Squared Error(MSE)
* Mean Absolute Error(MAE)
* R² Score

**Linear Regression Comparisons:**

**Error Metrics(MSE and MAE):** In 70% train set selected model, MSE and MAE values are less than the other model. It shows predictions errors are less than the model which 80% train set selected.

**R ² Score:** The R ² score of the model which 70% train set selected is bigger than the other model’s R ² score. It means the model which 70% train set selected is more successful than the other model.

***Conclusion***

The model trained with 30% test data performance is better than the other model.

**Random Forest Comparisons:**

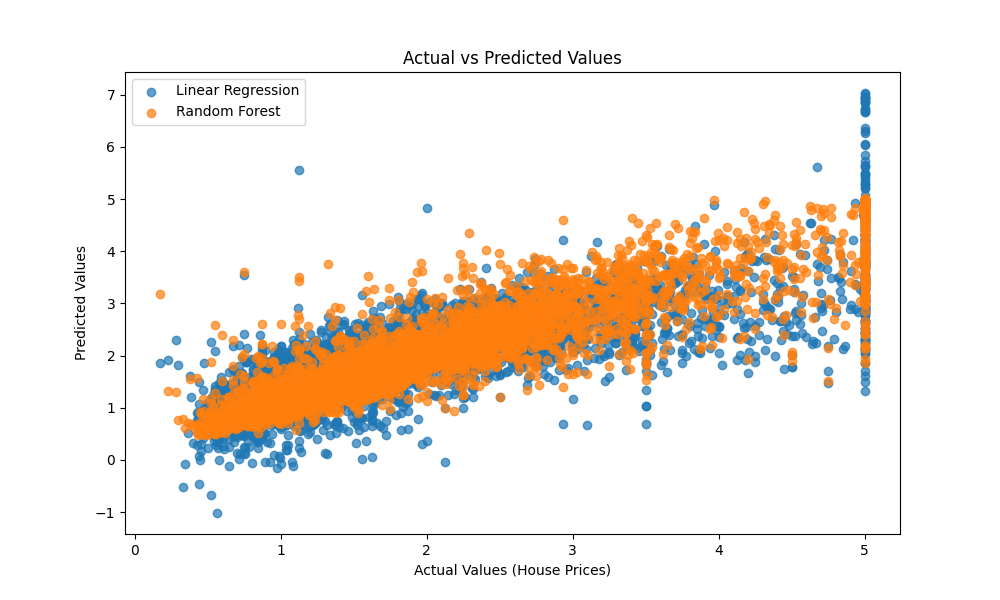
**Error Metrics(MSE and MAE):** In 80% train set selected model, MSE and MAE values are less than the other model. It shows predictions errors are less than the other model.

**R ² Score:** The R ² score of the model which 80% train set selected is bigger than the other model’s R ² score. It means the model which 80% train set selected is more successful than the other model.

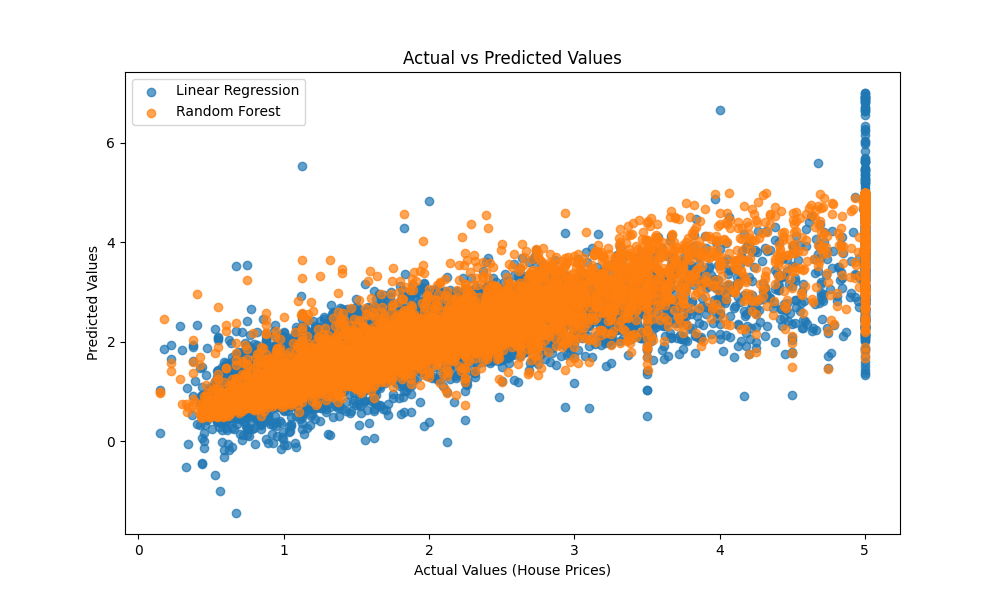
***Conclusion***

The model trained with 20% test data performance is better than the other model. This means that the model makes less errors and it explains the dependent variable better.

**Actual vs Predicted Values Scatter Plot for First Selection:**

****

**Actual vs Predicted Values Scatter Plot for Second Selection:**



* The random forest model performs better than linear regression on all metrics.
* Its lower MSE and MAE values indicate higher accuracy.

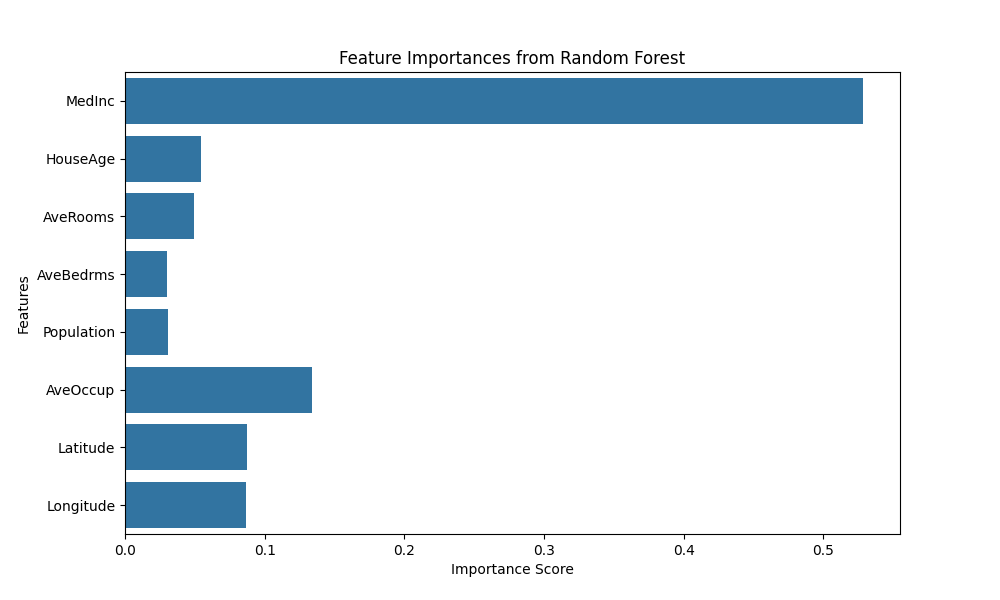
**5.5 Results and Discussion**

**5.5.1. Importance of Features**

The random forest model shows the importance of each feature in predicting house prices. Importances are the same in both models. The importance score is shown in the table below:

|  |  |
| --- | --- |
| **Feature** | **Importance Score** |
| MedInc | 0.52 |
| AveOccup | 0.05 |
| Latitude | 0.04 |
| Longitude | 0.02 |
| HouseAge | 0.03 |
| AveRooms | 0.13 |
| AveBedrms | 0.09 |
| Population | 0.08 |

**Feature Importance Bar Chart for Two Selections:**



**5.5.2. Model Comparison**

* **Lineer Regression:** Linear regression is easy to use and understand. It works well for simple data. But it does not work well with complex data. It is less accurate when the data is not linear.
* **Random Forest:** Random forest uses many decision trees to make better predictions. It works well with complex data and gives more accurate results than linear regression. It also shows which features are important in the data.

***Conclusion***

The random forest model is better for this task. It gives more accurate predictions and works well with complex data. It is a good choice for predicting house prices. It also helps to understand which features are important.

**6. Conclusion**

**6.1. Summary**

In this project, we used three main methods to work with data: classification, clustering, and regression. These methods helped us learn useful things from three different datasets.

The main points are shown below:

* **Classification**: We used decision tree and random forest models to predict if a person has heart disease. Random forest was better because it gave more accurate results and less mistakes.
* **Clustering**: The K-Means method grouped customers into three groups based on how much they spent on products. We used the elbow method to decide that three groups were the best choice.
* **Regression**: We predicted house prices using linear regression and random forest models. Random forest worked better because it understood the data better and made less errors.

**6.2. Analysis**

Each method worked well for its purpose. Here is why they are important:

* **Classification**: The random forest model was very good for predicting heart disease. It is reliable and can be used in medical studies to help doctors make better decisions.
* **Clustering**: The K-Means method can help businesses understand their customers. For example, they can see which customers spend a lot, which spend less, and then make better plans for these groups.
* **Regression**: The random forest model is strong and works well for complicated data like house prices. It can capture patterns that are not straight lines, making it better than simple models like linear regression.

**6.3. Recommendations**

Based on what we learned, here are some recommendations:

1. Use random forest when accuracy is very important, like in health or housing data.
2. Use clustering methods like K-Means to group similar data. This can help businesses understand their customers and improve their services.
3. Always check your data carefully before using any method. Make sure there are no missing values and scale the data if needed. Clean data gives better results.
4. Compare different methods to find the one that works best for your problem. For example, random forest was better than decision tree and linear regression in this project.

**6.4 Final Thoughts**

Data analysis is a powerful tool that helps us solve real-life problems. By finding patterns in data, we can make smarter decisions in fields like healthcare, business, and housing markets.

The methods we used in this project can help in many areas:

* In healthcare, they can predict diseases and improve patient care.
* In business, they can help companies understand customer behavior and make better plans.
* In housing, they can help people and companies predict house prices and make better investments.

Using the right method for the right data is very important. Random forest worked the best in this project, showing that advanced methods can give better results. With these tools, we can turn data into useful information and make better decisions in the future.