MSADS
Portfolio Milestone

Syracuse University, Spring 2024

Gianni C. Conde
SUID: 734859774
gcconde@syr.edu

Table of Contents

Introduction

The Applied Data Science Master's Degree program at Syracuse University supplies students with the expertise necessary to thrive in the analytically driven realm of data science. The emergence of various areas of study in the data science field led the program to design tracks to accommodate those pursuing careers in Artificial Intelligence, Business Analytics, Data Pipelines and Platforms, Language Analytics, Project Management, and Visual Analytics.

The expertise that this program bestows onto their successful students include:

- Data collection, storage, and accessibility through identification and leveraging of applicable technologies.
- Create obtainable insight from data pertaining to various industries while utilizing the full data science life cycle.
- Generate actionable insight with the application of predictive models and visualizations.
- Efficient use of programming languages such as Python, R, SQL, etc.
- Communication of analytical insights and visualizations to a broad range of stakeholders.
- Ethical development, use, and evaluation of data and models to ensure fairness, privacy, and bias.

These acquired skills are evident in the various projects that I have completed throughout the span of the program, particularly in courses such as IST 659: Data Administration Concepts & Database Management, IST 707: Applied Machine Learning, and IST 736: Text Mining. The code repository can be accessed in the link below:

https://github.com/gianniconde/IST-782---MSADS.git

IST 659 – Data Administration Concepts & Database Management

Course Description

This course provided hands-on involvement when defining, developing, and managing relational databases for information systems. Database application techniques were applied through the use of schema design, modeling, search specifications, and query languages. These techniques provided expertise on the database developmental life cycle, the construction of database objects through SQL, and the evaluation, improvement, and critique of database management systems.

Project Description

Goal: To create a database of free agent eligible Major League Baseball (MLB) players following the 2021 season to determine how their previous and projected seasons will impact their potential contracts with their future teams.

Contributors: Gianni Conde

About the Data: The dataset was retrieved from the 2022 MLB Free Agent Tracker page from Fangraphs. The data set includes 262 free agent eligible players, accompanied by attributes detailing their skillsets and their eventual contracts signed prior to the 2022 season. The top 20 free agents based on WAR for the 2021 season will be observed. The attributes include name, position, bats, throws, previous team, age, service time, 2021 Wins Above Replacement (WAR), projected 2022 WAR, Qualifying Offer (QO), signing team, years, total salary, and Average Annual Value (AAV). The two WAR attributes were the only statistical identifiers of player evaluation.

Glossary	
Player	Player who is a free agent
Position	Position(s) played
PlayerPOS	Bridge table indicating a player's position
Bats	Bat handedness
Throws	Throwing hand
Age	Age in years
ServiceTime	Years spent on the major league roster
PreviousTeamAbbr	Abbreviation of the player's team during the 2021 season
QO	Qualifying offer
WAR	Wins above replacement
WAR_2021	Player's WAR from the 2021 season
ProjWar_2022	Player's projected WAR for the 2022 season
NewContract	Player's contract signed beginning in the 2022 season
SigningTeamAbbr	Abbreviation of the team that signs a player for the 2022 season
TotalSalary	Total salary in dollars
Years	Length of the contract in years
AAV	Average annual salary (TotalSalary / Years) in dollars

Business Rules:

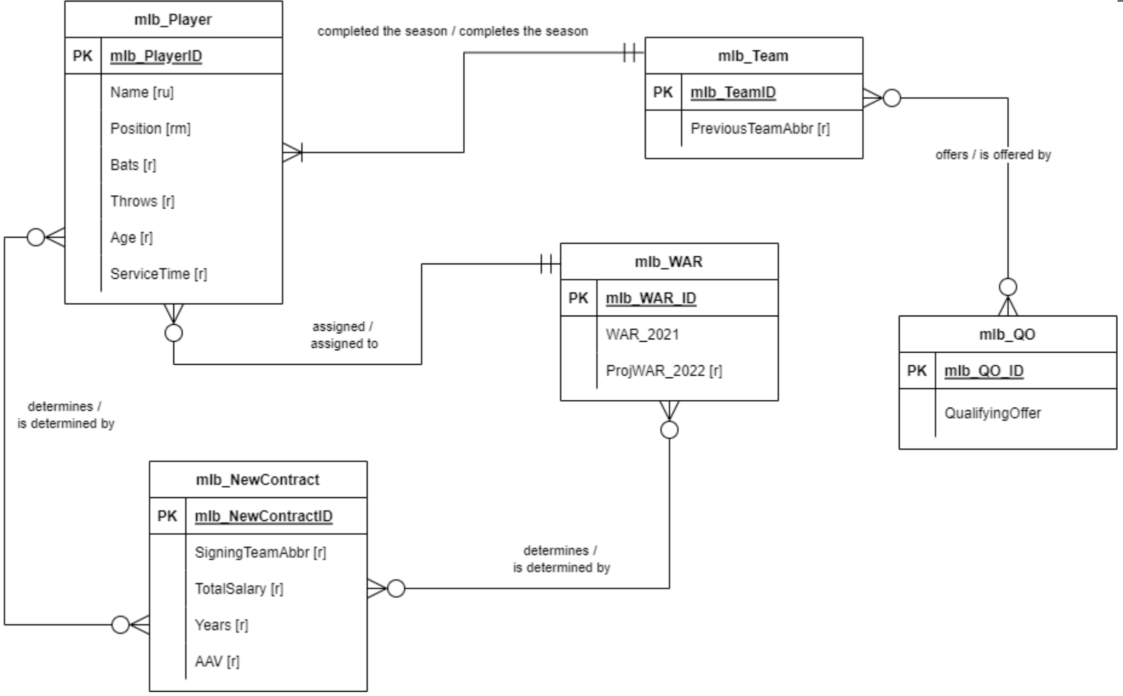
- Every attribute in the data set is required except for suffix, qualifying offer (QO), and 2021 WAR.
- Players who have been traded to new teams during the 2021 season prior to free agency are not eligible to receive a qualifying offer. Since the data does not include multiple previous teams, each player who did not receive one will have an empty space for this attribute.
- Players who did not play during the 2021 season due to injury or taking their services abroad will not have a value for 2021 WAR.
- A player must play at least one position.
- Age, service time, QO, years, total salary, and AAV cannot be negative.

- 2021 WAR and projected 2022 WAR can be negative.
- Previous teams and signing teams are not multivalued.

Step 1: Entity Relationship Diagram (ERD)

Prior to creating the ERD, a visualization of a list of entities, attributes, and relationships is provided.

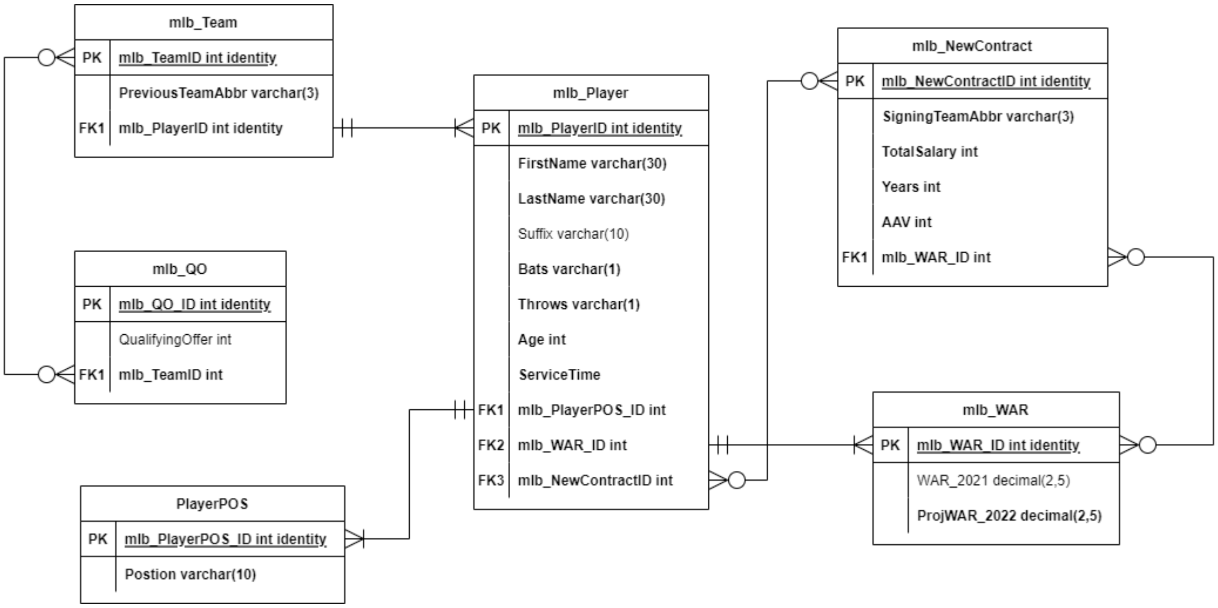
Entity	**Attribute**
mlb_Player	Name [ru], Position [rm], Bats [r], Throws [r], Age [r], ServiceTime [r]
mlb_Team	PreviousTeamAbbr [r]
mlb_QO	QualifyingOffer
mlb_WAR	WAR_2021, ProjWAR_2022
mlb_NewContract	SigningTeamAbbr [r], TotalSalary [r], Years [r], AAV [r]
Relationships	
o Each free agent eligible player completed the 2021 season with exactly one team; each team completes the season with one or more free agent eligible players. o Each team offers zero or more qualifying offers; each qualifying offer is offered by zero or more teams. o Each player is assigned exactly one value of WAR per season; each value of WAR is assigned to all players. o Each player's mlb_WAR determines their mlb_NewContract's total salary; each mlb_NewContract's total salary is determined by a player's mlb_WAR. o Each player's age determines their mlb_NewContract's years; each mlb_NewContract's years is determined by a player's age.	



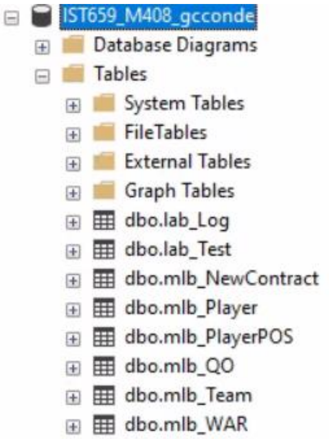
Step 2: Logical Model

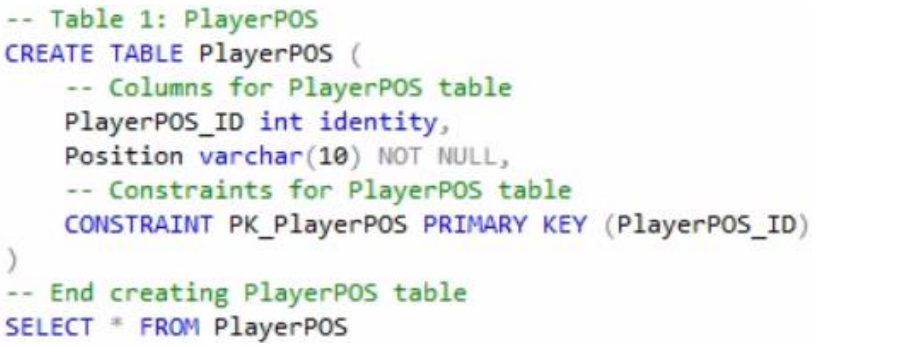
Normalization of the model is necessary so that it will be in third nominal form (3NF). To be in 3NF, a model must first be in 1NF, then 2NF. Since the conceptual model has a multivalued attribute (Position [rm]), it is currently in 0NF. A new table titled PlayerPosition was created and the multivalued attribute Position was transposed, thus converting the model to 1NF.

To convert the model to 2NF, the candidate keys were identified as Position, 2021_WAR, and TotalSalary. PFDs were removed by creating new relations in their place. Lastly, to convert the model to 3NF, all TFDs were resolved by creating new relations with FKs.

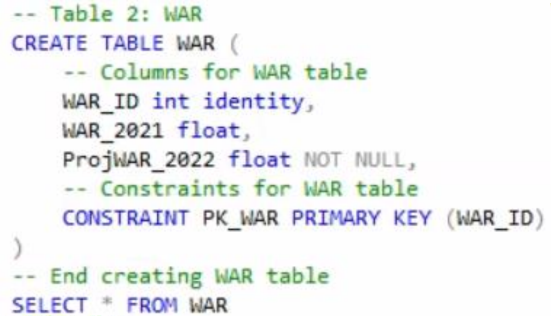


<u>Step 3</u>: Database Design

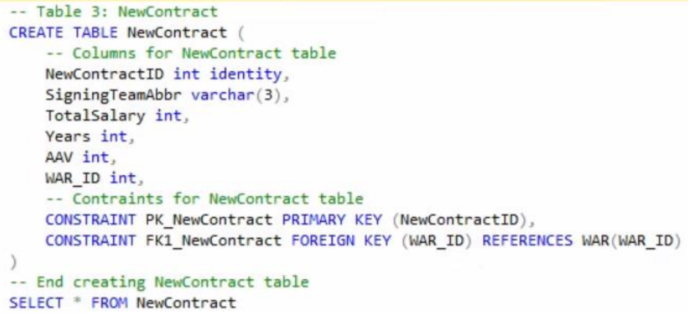




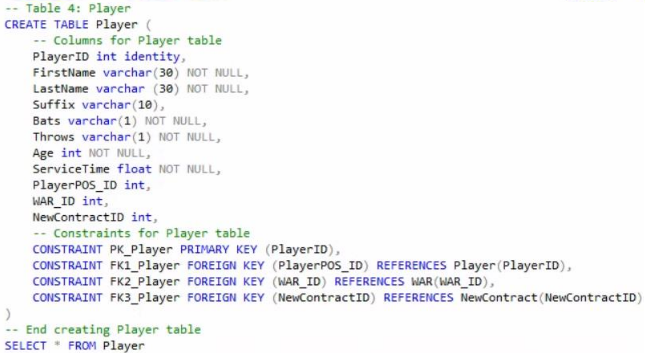
```
-- Table 1: PlayerPOS
CREATE TABLE PlayerPOS (
    -- Columns for PlayerPOS table
    PlayerPOS_ID int identity,
    Position varchar(10) NOT NULL,
    -- Constraints for PlayerPOS table
    CONSTRAINT PK_PlayerPOS PRIMARY KEY (PlayerPOS_ID)
)
-- End creating PlayerPOS table
SELECT * FROM PlayerPOS
```



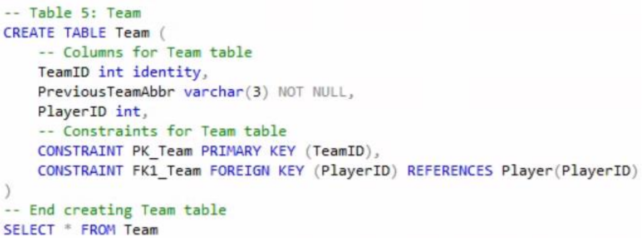
```
-- Table 2: WAR
CREATE TABLE WAR (
    -- Columns for WAR table
    WAR_ID int identity,
    WAR_2021 float,
    ProjWAR_2022 float NOT NULL,
    -- Constraints for WAR table
    CONSTRAINT PK_WAR PRIMARY KEY (WAR_ID)
)
-- End creating WAR table
SELECT * FROM WAR
```



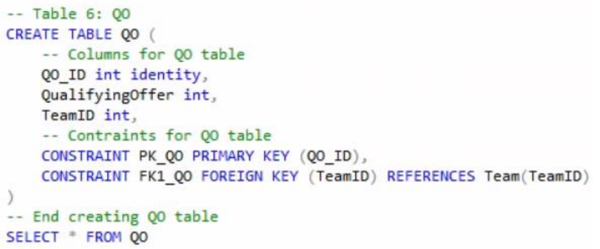
```
-- Table 3: NewContract
CREATE TABLE NewContract (
    -- Columns for NewContract table
    NewContractID int identity,
    SigningTeamAbbr varchar(3),
    TotalSalary int,
    Years int,
    AAV int,
    WAR_ID int,
    -- Contraints for NewContract table
    CONSTRAINT PK_NewContract PRIMARY KEY (NewContractID),
    CONSTRAINT FK1_NewContract FOREIGN KEY (WAR_ID) REFERENCES WAR(WAR_ID)
)
-- End creating NewContract table
SELECT * FROM NewContract
```



```
-- Table 4: Player
CREATE TABLE Player (
    -- Columns for Player table
    PlayerID int identity,
    FirstName varchar(30) NOT NULL,
    LastName varchar (30) NOT NULL,
    Suffix varchar(10),
    Bats varchar(1) NOT NULL,
    Throws varchar(1) NOT NULL,
    Age int NOT NULL,
    ServiceTime float NOT NULL,
    PlayerPOS_ID int,
    WAR_ID int,
    NewContractID int,
    -- Constraints for Player table
    CONSTRAINT PK_Player PRIMARY KEY (PlayerID),
    CONSTRAINT FK1_Player FOREIGN KEY (PlayerPOS_ID) REFERENCES Player(PlayerID),
    CONSTRAINT FK2_Player FOREIGN KEY (WAR_ID) REFERENCES WAR(WAR_ID),
    CONSTRAINT FK3_Player FOREIGN KEY (NewContractID) REFERENCES NewContract(NewContractID)
)
-- End creating Player table
SELECT * FROM Player
```



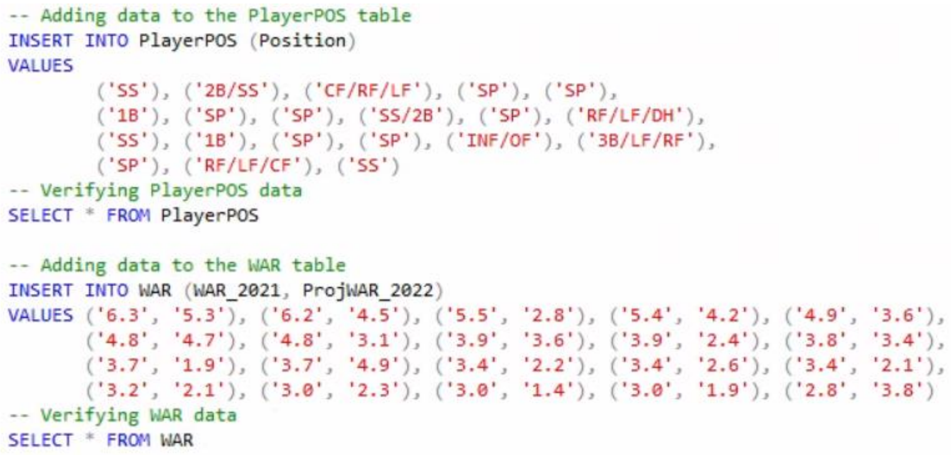
```
-- Table 5: Team
CREATE TABLE Team (
    -- Columns for Team table
    TeamID int identity,
    PreviousTeamAbbr varchar(3) NOT NULL,
    PlayerID int,
    -- Constraints for Team table
    CONSTRAINT PK_Team PRIMARY KEY (TeamID),
    CONSTRAINT FK1_Team FOREIGN KEY (PlayerID) REFERENCES Player(PlayerID)
)
-- End creating Team table
SELECT * FROM Team
```



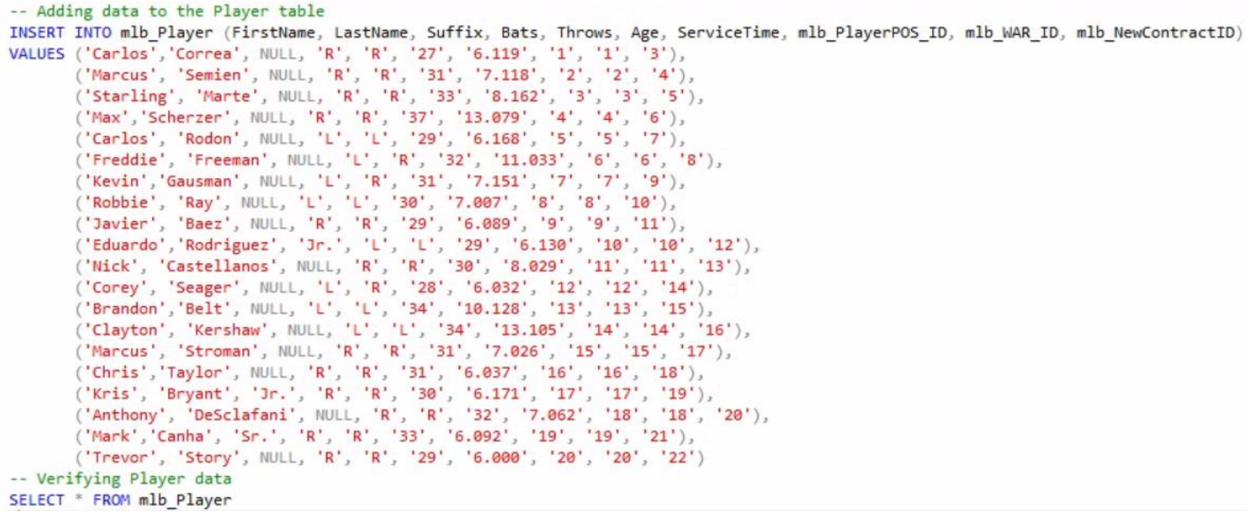
```
-- Table 6: QO
CREATE TABLE QO (
    -- Columns for QO table
    QO_ID int identity,
    QualifyingOffer int,
    TeamID int,
    -- Contraints for QO table
    CONSTRAINT PK_QO PRIMARY KEY (QO_ID),
    CONSTRAINT FK1_QO FOREIGN KEY (TeamID) REFERENCES Team(TeamID)
)
-- End creating QO table
SELECT * FROM QO
```

Queries were created to build six tables representing player position, WAR, new contracts, players, teams, and QO.

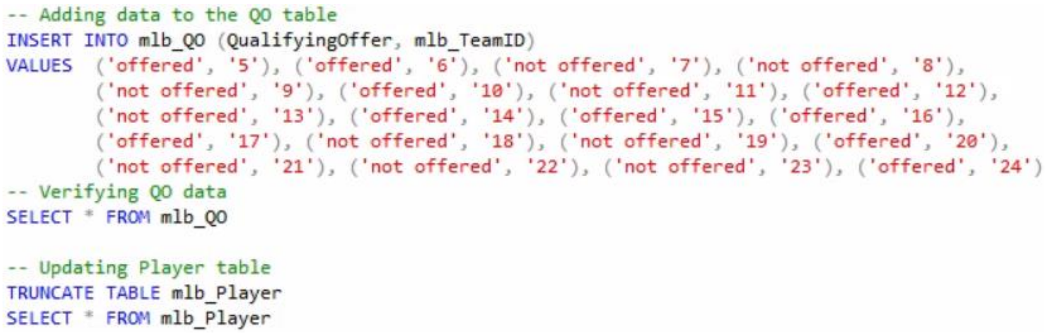
Step 4: Data Creation



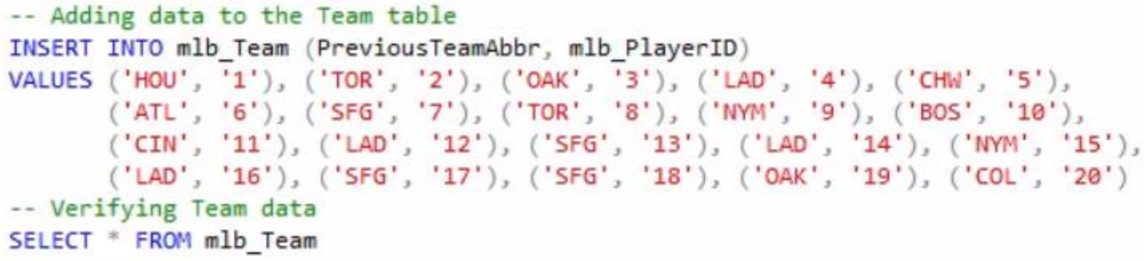
```
-- Adding data to the PlayerPOS table
INSERT INTO PlayerPOS (Position)
VALUES
        ('SS'), ('2B/SS'), ('CF/RF/LF'), ('SP'), ('SP'),
        ('1B'), ('SP'), ('SP'), ('SS/2B'), ('SP'), ('RF/LF/DH'),
        ('SS'), ('1B'), ('SP'), ('SP'), ('INF/OF'), ('3B/LF/RF'),
        ('SP'), ('RF/LF/CF'), ('SS')
-- Verifying PlayerPOS data
SELECT * FROM PlayerPOS

-- Adding data to the WAR table
INSERT INTO WAR (WAR_2021, ProjWAR_2022)
VALUES ('6.3', '5.3'), ('6.2', '4.5'), ('5.5', '2.8'), ('5.4', '4.2'), ('4.9', '3.6'),
       ('4.8', '4.7'), ('4.8', '3.1'), ('3.9', '3.6'), ('3.9', '2.4'), ('3.8', '3.4'),
       ('3.7', '1.9'), ('3.7', '4.9'), ('3.4', '2.2'), ('3.4', '2.6'), ('3.4', '2.1'),
       ('3.2', '2.1'), ('3.0', '2.3'), ('3.0', '1.4'), ('3.0', '1.9'), ('2.8', '3.8')
-- Verifying WAR data
SELECT * FROM WAR
```



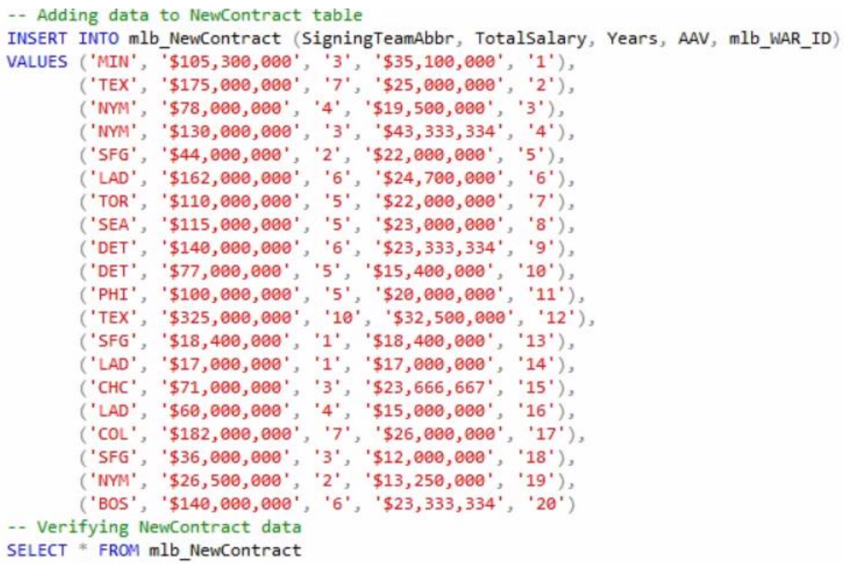
```
-- Adding data to the Player table
INSERT INTO mlb_Player (FirstName, LastName, Suffix, Bats, Throws, Age, ServiceTime, mlb_PlayerPOS_ID, mlb_WAR_ID, mlb_NewContractID)
VALUES ('Carlos','Correa', NULL, 'R', 'R', '27', '6.119', '1', '1', '3'),
       ('Marcus', 'Semien', NULL, 'R', 'R', '31', '7.118', '2', '2', '4'),
       ('Starling', 'Marte', NULL, 'R', 'R', '33', '8.162', '3', '3', '5'),
       ('Max','Scherzer', NULL, 'R', 'R', '37', '13.079', '4', '4', '6'),
       ('Carlos', 'Rodon', NULL, 'L', 'L', '29', '6.168', '5', '5', '7'),
       ('Freddie', 'Freeman', NULL, 'L', 'R', '32', '11.033', '6', '6', '8'),
       ('Kevin','Gausman', NULL, 'L', 'R', '31', '7.151', '7', '7', '9'),
       ('Robbie', 'Ray', NULL, 'L', 'L', '30', '7.007', '8', '8', '10'),
       ('Javier', 'Baez', NULL, 'R', 'R', '29', '6.089', '9', '9', '11'),
       ('Eduardo','Rodriguez', 'Jr.', 'L', 'L', '29', '6.130', '10', '10', '12'),
       ('Nick', 'Castellanos', NULL, 'R', 'R', '30', '8.029', '11', '11', '13'),
       ('Corey', 'Seager', NULL, 'L', 'R', '28', '6.032', '12', '12', '14'),
       ('Brandon','Belt', NULL, 'L', 'L', '34', '10.128', '13', '13', '15'),
       ('Clayton', 'Kershaw', NULL, 'L', 'L', '34', '13.105', '14', '14', '16'),
       ('Marcus', 'Stroman', NULL, 'R', 'R', '31', '7.026', '15', '15', '17'),
       ('Chris','Taylor', NULL, 'R', 'R', '31', '6.037', '16', '16', '18'),
       ('Kris', 'Bryant', 'Jr.', 'R', 'R', '30', '6.171', '17', '17', '19'),
       ('Anthony', 'DeSclafani', NULL, 'R', 'R', '32', '7.062', '18', '18', '20'),
       ('Mark','Canha', 'Sr.', 'R', 'R', '33', '6.092', '19', '19', '21'),
       ('Trevor', 'Story', NULL, 'R', 'R', '29', '6.000', '20', '20', '22')
-- Verifying Player data
SELECT * FROM mlb_Player
```



```
-- Adding data to the QO table
INSERT INTO mlb_QO (QualifyingOffer, mlb_TeamID)
VALUES  ('offered', '5'), ('offered', '6'), ('not offered', '7'), ('not offered', '8'),
        ('not offered', '9'), ('offered', '10'), ('not offered', '11'), ('offered', '12'),
        ('not offered', '13'), ('offered', '14'), ('offered', '15'), ('offered', '16'),
        ('offered', '17'), ('not offered', '18'), ('not offered', '19'), ('offered', '20'),
        ('not offered', '21'), ('not offered', '22'), ('not offered', '23'), ('offered', '24')
-- Verifying QO data
SELECT * FROM mlb_QO

-- Updating Player table
TRUNCATE TABLE mlb_Player
SELECT * FROM mlb_Player
```



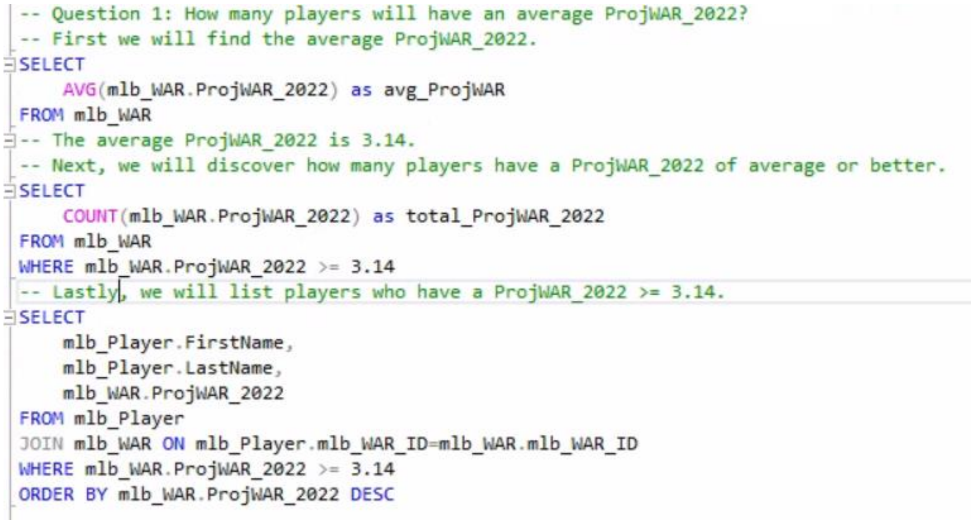
```
-- Adding data to the Team table
INSERT INTO mlb_Team (PreviousTeamAbbr, mlb_PlayerID)
VALUES ('HOU', '1'), ('TOR', '2'), ('OAK', '3'), ('LAD', '4'), ('CHW', '5'),
       ('ATL', '6'), ('SFG', '7'), ('TOR', '8'), ('NYM', '9'), ('BOS', '10'),
       ('CIN', '11'), ('LAD', '12'), ('SFG', '13'), ('LAD', '14'), ('NYM', '15'),
       ('LAD', '16'), ('SFG', '17'), ('SFG', '18'), ('OAK', '19'), ('COL', '20')
-- Verifying Team data
SELECT * FROM mlb_Team
```



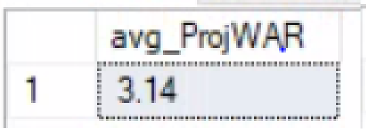
```
-- Adding data to NewContract table
INSERT INTO mlb_NewContract (SigningTeamAbbr, TotalSalary, Years, AAV, mlb_WAR_ID)
VALUES ('MIN', '$105,300,000', '3', '$35,100,000', '1'),
       ('TEX', '$175,000,000', '7', '$25,000,000', '2'),
       ('NYM', '$78,000,000', '4', '$19,500,000', '3'),
       ('NYM', '$130,000,000', '3', '$43,333,334', '4'),
       ('SFG', '$44,000,000', '2', '$22,000,000', '5'),
       ('LAD', '$162,000,000', '6', '$24,700,000', '6'),
       ('TOR', '$110,000,000', '5', '$22,000,000', '7'),
       ('SEA', '$115,000,000', '5', '$23,000,000', '8'),
       ('DET', '$140,000,000', '6', '$23,333,334', '9'),
       ('DET', '$77,000,000', '5', '$15,400,000', '10'),
       ('PHI', '$100,000,000', '5', '$20,000,000', '11'),
       ('TEX', '$325,000,000', '10', '$32,500,000', '12'),
       ('SFG', '$18,400,000', '1', '$18,400,000', '13'),
       ('LAD', '$17,000,000', '1', '$17,000,000', '14'),
       ('CHC', '$71,000,000', '3', '$23,666,667', '15'),
       ('LAD', '$60,000,000', '4', '$15,000,000', '16'),
       ('COL', '$182,000,000', '7', '$26,000,000', '17'),
       ('SFG', '$36,000,000', '3', '$12,000,000', '18'),
       ('NYM', '$26,500,000', '2', '$13,250,000', '19'),
       ('BOS', '$140,000,000', '6', '$23,333,334', '20')
-- Verifying NewContract data
SELECT * FROM mlb_NewContract
```

The actual data was created my adding them to their appropriate tables.

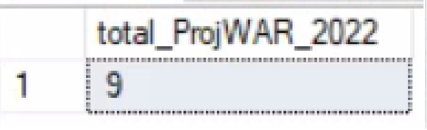
Step 5: How many players will have an average ProjWAR_2022?



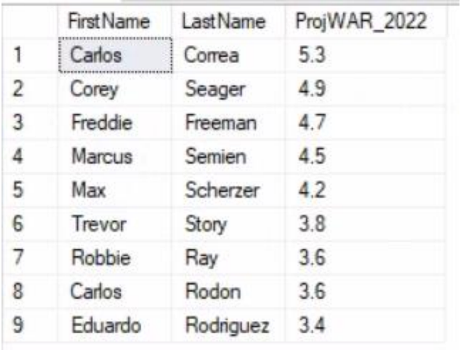
```
-- Question 1: How many players will have an average ProjWAR_2022?
-- First we will find the average ProjWAR_2022.
SELECT
    AVG(mlb_WAR.ProjWAR_2022) as avg_ProjWAR
FROM mlb_WAR
-- The average ProjWAR_2022 is 3.14.
-- Next, we will discover how many players have a ProjWAR_2022 of average or better.
SELECT
    COUNT(mlb_WAR.ProjWAR_2022) as total_ProjWAR_2022
FROM mlb_WAR
WHERE mlb_WAR.ProjWAR_2022 >= 3.14
-- Lastly, we will list players who have a ProjWAR_2022 >= 3.14.
SELECT
    mlb_Player.FirstName,
    mlb_Player.LastName,
    mlb_WAR.ProjWAR_2022
FROM mlb_Player
JOIN mlb_WAR ON mlb_Player.mlb_WAR_ID=mlb_WAR.mlb_WAR_ID
WHERE mlb_WAR.ProjWAR_2022 >= 3.14
ORDER BY mlb_WAR.ProjWAR_2022 DESC
```



	avg_ProjWAR
1	3.14



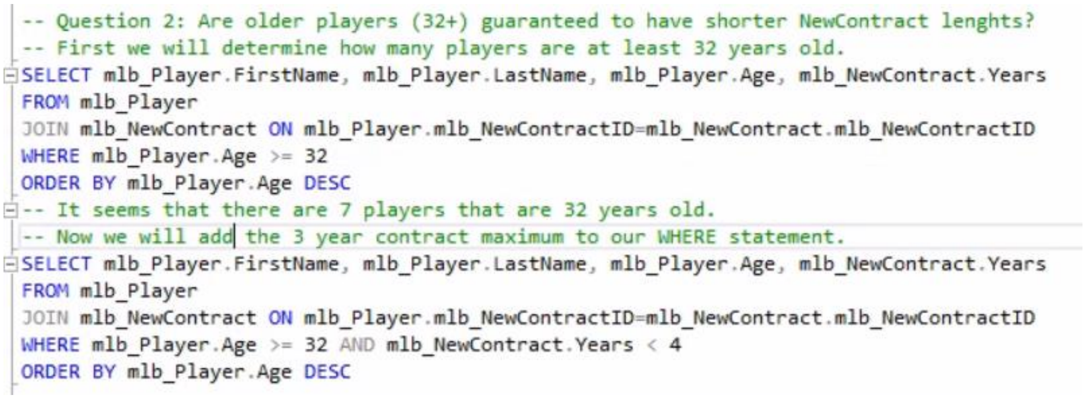
	total_ProjWAR_2022
1	9



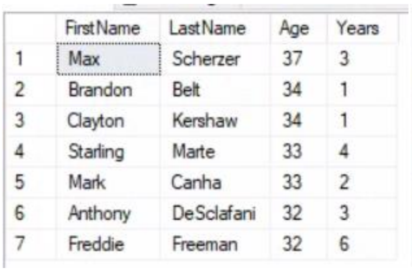
	FirstName	LastName	ProjWAR_2022
1	Carlos	Correa	5.3
2	Corey	Seager	4.9
3	Freddie	Freeman	4.7
4	Marcus	Semien	4.5
5	Max	Scherzer	4.2
6	Trevor	Story	3.8
7	Robbie	Ray	3.6
8	Carlos	Rodon	3.6
9	Eduardo	Rodriguez	3.4

The average projected WAR for the 2022 season is 3.14. Nine total players were projected to have a projected WAR equal to or greater than the projected average.

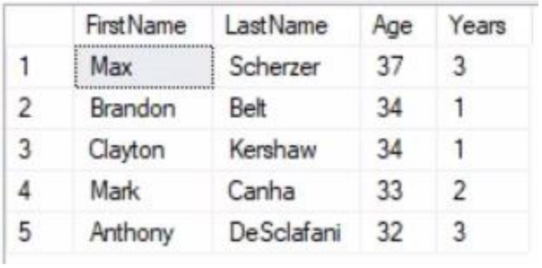
<u>Step 6</u>: Are older players (32+) guaranteed to have shorter contract lengths (less than 4 years)?



```
-- Question 2: Are older players (32+) guaranteed to have shorter NewContract lenghts?
-- First we will determine how many players are at least 32 years old.
SELECT mlb_Player.FirstName, mlb_Player.LastName, mlb_Player.Age, mlb_NewContract.Years
FROM mlb_Player
JOIN mlb_NewContract ON mlb_Player.mlb_NewContractID=mlb_NewContract.mlb_NewContractID
WHERE mlb_Player.Age >= 32
ORDER BY mlb_Player.Age DESC
-- It seems that there are 7 players that are 32 years old.
-- Now we will add the 3 year contract maximum to our WHERE statement.
SELECT mlb_Player.FirstName, mlb_Player.LastName, mlb_Player.Age, mlb_NewContract.Years
FROM mlb_Player
JOIN mlb_NewContract ON mlb_Player.mlb_NewContractID=mlb_NewContract.mlb_NewContractID
WHERE mlb_Player.Age >= 32 AND mlb_NewContract.Years < 4
ORDER BY mlb_Player.Age DESC
```



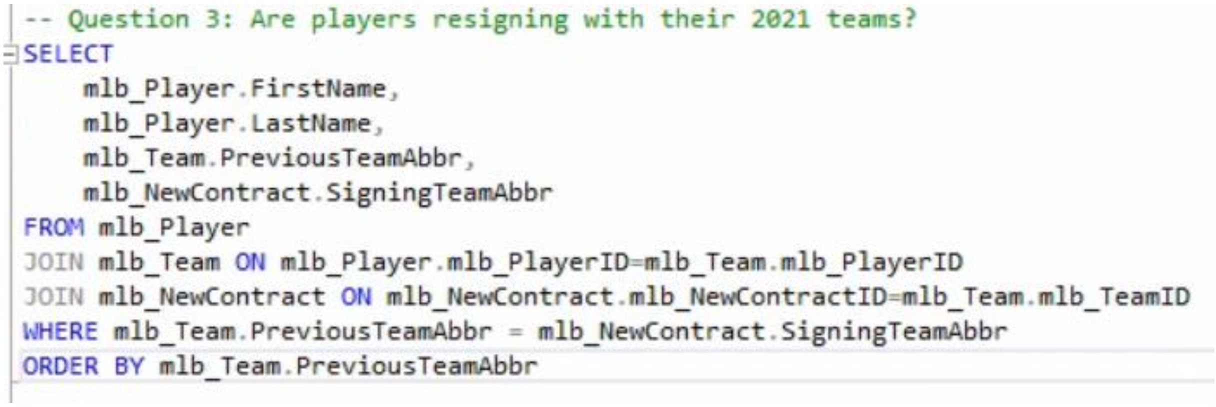
	FirstName	LastName	Age	Years
1	Max	Scherzer	37	3
2	Brandon	Belt	34	1
3	Clayton	Kershaw	34	1
4	Starling	Marte	33	4
5	Mark	Canha	33	2
6	Anthony	DeSclafani	32	3
7	Freddie	Freeman	32	6



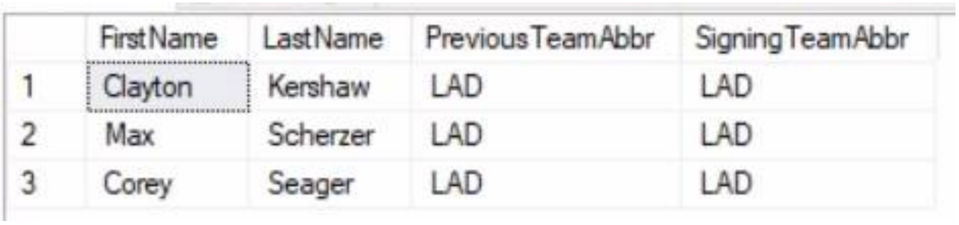
	FirstName	LastName	Age	Years
1	Max	Scherzer	37	3
2	Brandon	Belt	34	1
3	Clayton	Kershaw	34	1
4	Mark	Canha	33	2
5	Anthony	DeSclafani	32	3

Of the seven players 32 years and older, five of them signed new contracts of less than 4 years. Since contracts of less than 4 years are considered to be short-term, we know that older players may not be guaranteed to sign shorter contracts but are more likely to.

<u>Step 7</u>: Are players resigning with their 2021 teams?



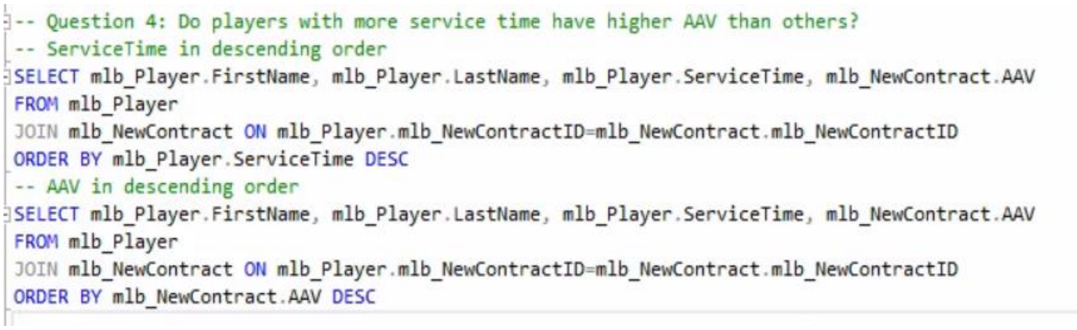
```
-- Question 3: Are players resigning with their 2021 teams?
SELECT
    mlb_Player.FirstName,
    mlb_Player.LastName,
    mlb_Team.PreviousTeamAbbr,
    mlb_NewContract.SigningTeamAbbr
FROM mlb_Player
JOIN mlb_Team ON mlb_Player.mlb_PlayerID=mlb_Team.mlb_PlayerID
JOIN mlb_NewContract ON mlb_NewContract.mlb_NewContractID=mlb_Team.mlb_TeamID
WHERE mlb_Team.PreviousTeamAbbr = mlb_NewContract.SigningTeamAbbr
ORDER BY mlb_Team.PreviousTeamAbbr
```



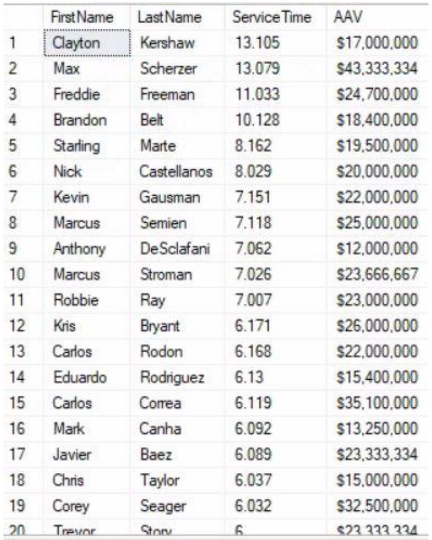
	FirstName	LastName	PreviousTeamAbbr	SigningTeamAbbr
1	Clayton	Kershaw	LAD	LAD
2	Max	Scherzer	LAD	LAD
3	Corey	Seager	LAD	LAD

By joining the mlb_Player, mlb_Team, and mlb_NewContract tables, there appears to only be three players who resigned with their former teams.

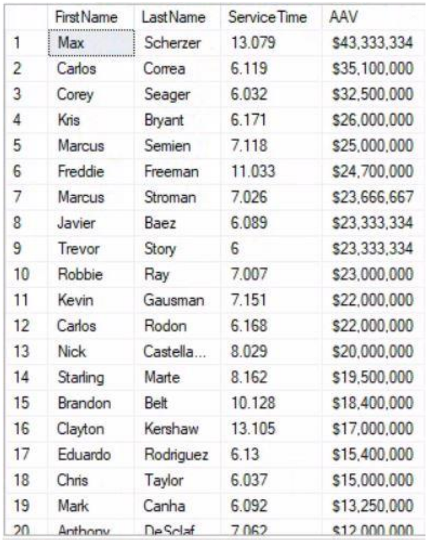
Step 8: Do players with more service time have higher AAV than others?



```
-- Question 4: Do players with more service time have higher AAV than others?
-- ServiceTime in descending order
SELECT mlb_Player.FirstName, mlb_Player.LastName, mlb_Player.ServiceTime, mlb_NewContract.AAV
FROM mlb_Player
JOIN mlb_NewContract ON mlb_Player.mlb_NewContractID=mlb_NewContract.mlb_NewContractID
ORDER BY mlb_Player.ServiceTime DESC
-- AAV in descending order
SELECT mlb_Player.FirstName, mlb_Player.LastName, mlb_Player.ServiceTime, mlb_NewContract.AAV
FROM mlb_Player
JOIN mlb_NewContract ON mlb_Player.mlb_NewContractID=mlb_NewContract.mlb_NewContractID
ORDER BY mlb_NewContract.AAV DESC
```



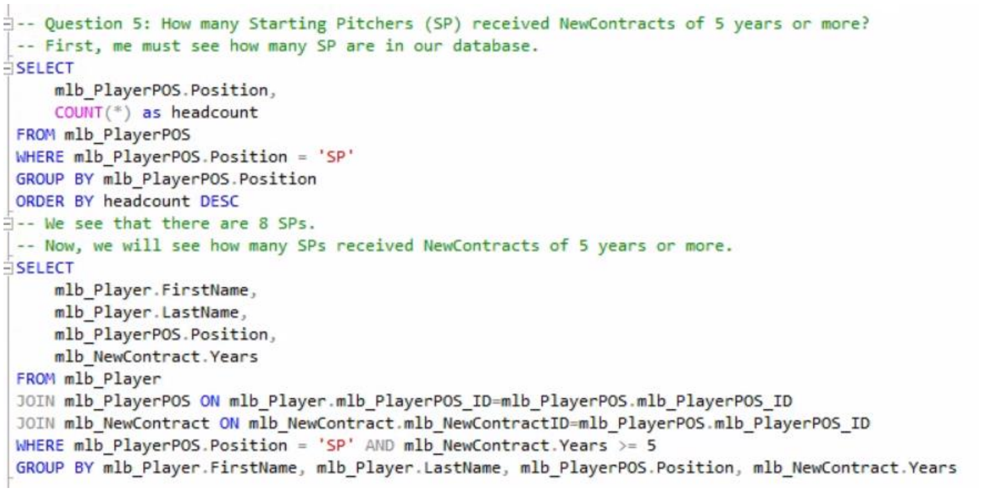
	FirstName	LastName	ServiceTime	AAV
1	Clayton	Kershaw	13.105	$17,000,000
2	Max	Scherzer	13.079	$43,333,334
3	Freddie	Freeman	11.033	$24,700,000
4	Brandon	Belt	10.128	$18,400,000
5	Starling	Marte	8.162	$19,500,000
6	Nick	Castellanos	8.029	$20,000,000
7	Kevin	Gausman	7.151	$22,000,000
8	Marcus	Semien	7.118	$25,000,000
9	Anthony	DeSclafani	7.062	$12,000,000
10	Marcus	Stroman	7.026	$23,666,667
11	Robbie	Ray	7.007	$23,000,000
12	Kris	Bryant	6.171	$26,000,000
13	Carlos	Rodon	6.168	$22,000,000
14	Eduardo	Rodriguez	6.13	$15,400,000
15	Carlos	Correa	6.119	$35,100,000
16	Mark	Canha	6.092	$13,250,000
17	Javier	Baez	6.089	$23,333,334
18	Chris	Taylor	6.037	$15,000,000
19	Corey	Seager	6.032	$32,500,000
20	Trevor	Story	6	$23,333,334



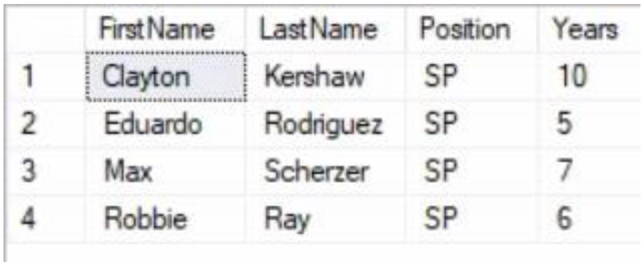
	FirstName	LastName	ServiceTime	AAV
1	Max	Scherzer	13.079	$43,333,334
2	Carlos	Correa	6.119	$35,100,000
3	Corey	Seager	6.032	$32,500,000
4	Kris	Bryant	6.171	$26,000,000
5	Marcus	Semien	7.118	$25,000,000
6	Freddie	Freeman	11.033	$24,700,000
7	Marcus	Stroman	7.026	$23,666,667
8	Javier	Baez	6.089	$23,333,334
9	Trevor	Story	6	$23,333,334
10	Robbie	Ray	7.007	$23,000,000
11	Kevin	Gausman	7.151	$22,000,000
12	Carlos	Rodon	6.168	$22,000,000
13	Nick	Castella...	8.029	$20,000,000
14	Starling	Marte	8.162	$19,500,000
15	Brandon	Belt	10.128	$18,400,000
16	Clayton	Kershaw	13.105	$17,000,000
17	Eduardo	Rodriguez	6.13	$15,400,000
18	Chris	Taylor	6.037	$15,000,000
19	Mark	Canha	6.092	$13,250,000
20	Anthony	DeSclaf	7.062	$12,000,000

By joining the mlb_Player and mlb_NewContract tables, it appears that Players with more ServiceTime do not necessarily have higher AAVs in their NewContracts. For example, while Clayton Kershaw has the second highest ServiceTime (13.105) and the 16th highest AAV ($17,000,000), Freddie Freeman has the third highest ServiceTime (11.033) and the sixth highest AAV ($24,700,000) which is 10 spots higher than Kershaw's.

Step 9: How many Starting Pitchers (SP) received new contracts of 5 years or more?



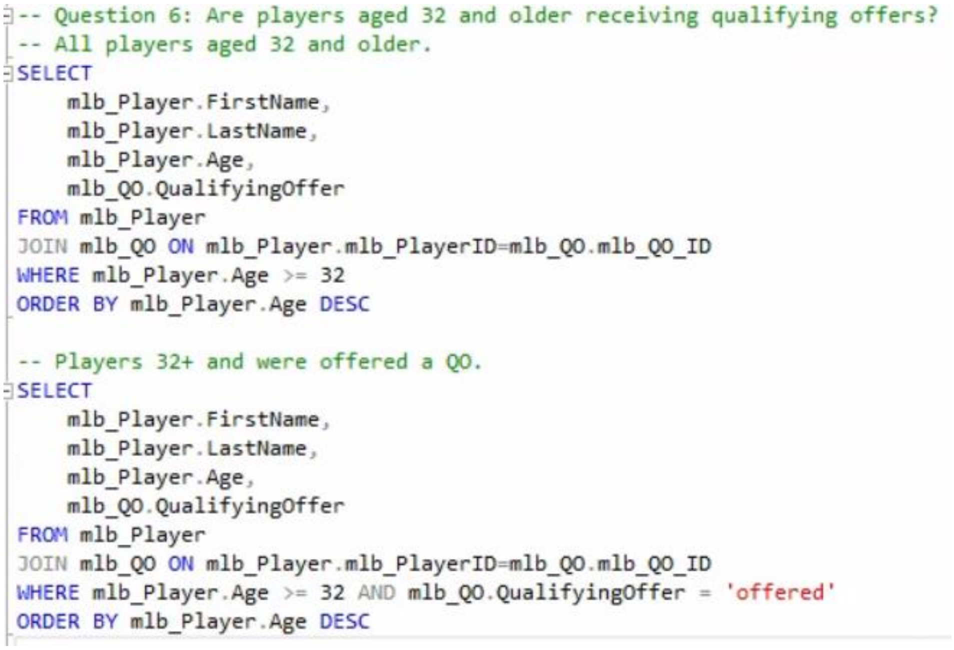
```
-- Question 5: How many Starting Pitchers (SP) received NewContracts of 5 years or more?
-- First, me must see how many SP are in our database.
SELECT
    mlb_PlayerPOS.Position,
    COUNT(*) as headcount
FROM mlb_PlayerPOS
WHERE mlb_PlayerPOS.Position = 'SP'
GROUP BY mlb_PlayerPOS.Position
ORDER BY headcount DESC
-- We see that there are 8 SPs.
-- Now, we will see how many SPs received NewContracts of 5 years or more.
SELECT
    mlb_Player.FirstName,
    mlb_Player.LastName,
    mlb_PlayerPOS.Position,
    mlb_NewContract.Years
FROM mlb_Player
JOIN mlb_PlayerPOS ON mlb_Player.mlb_PlayerPOS_ID=mlb_PlayerPOS.mlb_PlayerPOS_ID
JOIN mlb_NewContract ON mlb_NewContract.mlb_NewContractID=mlb_PlayerPOS.mlb_PlayerPOS_ID
WHERE mlb_PlayerPOS.Position = 'SP' AND mlb_NewContract.Years >= 5
GROUP BY mlb_Player.FirstName, mlb_Player.LastName, mlb_PlayerPOS.Position, mlb_NewContract.Years
```



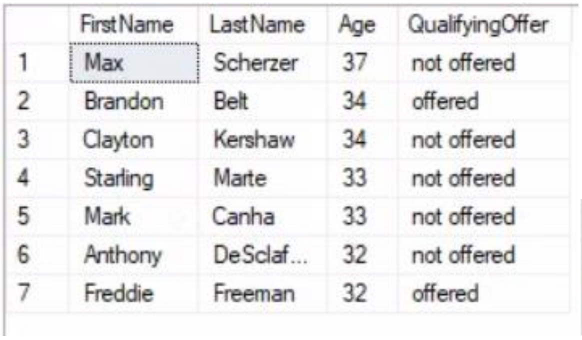
	FirstName	LastName	Position	Years
1	Clayton	Kershaw	SP	10
2	Eduardo	Rodriguez	SP	5
3	Max	Scherzer	SP	7
4	Robbie	Ray	SP	6

Four players received contracts of at least five years.

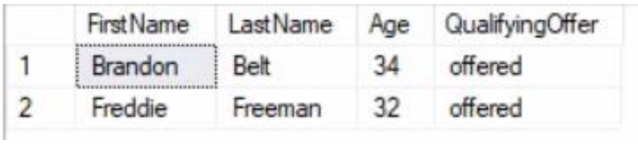
Step 10: Are players aged 32 and older receiving qualifying offers?



```
-- Question 6: Are players aged 32 and older receiving qualifying offers?
-- All players aged 32 and older.
SELECT
    mlb_Player.FirstName,
    mlb_Player.LastName,
    mlb_Player.Age,
    mlb_QO.QualifyingOffer
FROM mlb_Player
JOIN mlb_QO ON mlb_Player.mlb_PlayerID=mlb_QO.mlb_QO_ID
WHERE mlb_Player.Age >= 32
ORDER BY mlb_Player.Age DESC

-- Players 32+ and were offered a QO.
SELECT
    mlb_Player.FirstName,
    mlb_Player.LastName,
    mlb_Player.Age,
    mlb_QO.QualifyingOffer
FROM mlb_Player
JOIN mlb_QO ON mlb_Player.mlb_PlayerID=mlb_QO.mlb_QO_ID
WHERE mlb_Player.Age >= 32 AND mlb_QO.QualifyingOffer = 'offered'
ORDER BY mlb_Player.Age DESC
```



	FirstName	LastName	Age	QualifyingOffer
1	Max	Scherzer	37	not offered
2	Brandon	Belt	34	offered
3	Clayton	Kershaw	34	not offered
4	Starling	Marte	33	not offered
5	Mark	Canha	33	not offered
6	Anthony	DeSclaf...	32	not offered
7	Freddie	Freeman	32	offered

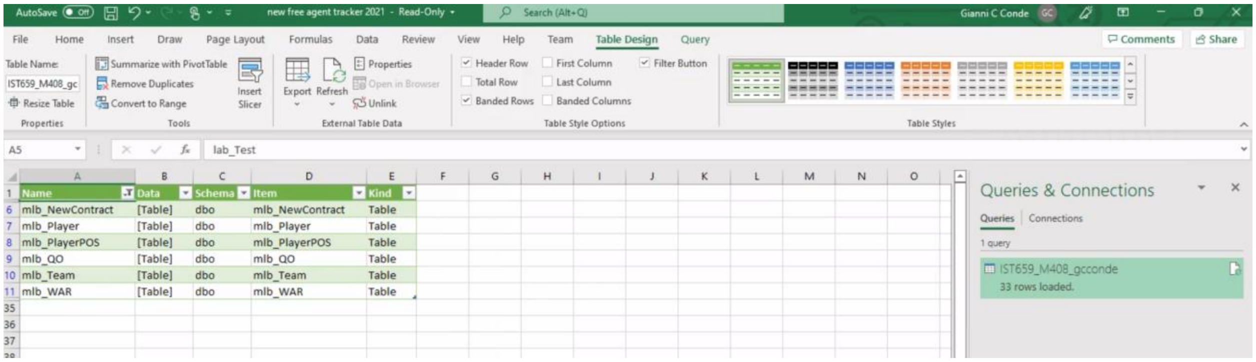


	FirstName	LastName	Age	QualifyingOffer
1	Brandon	Belt	34	offered
2	Freddie	Freeman	32	offered

Of the seven free agents that were at least 32 years old and eligible to receive qualifying offers, only two received them. Qualifying offers are essentially one-year deals whose values are set by arbitrators on a yearly basis. They also come with the incentive of the player's former team receiving a compensation draft pick from the player's new team. Qualifying offers are meant to not only retain players who feel as though they could produce a better season to potentially improve their future earnings, but to also deter teams from signing one of their star free agents with the hopes of the former team resigning them.

Step 11: Implementation

Microsoft Excel was the chosen tool for building a front-end, providing user interface for maintenance and query reporting.



A5 | lab_Test

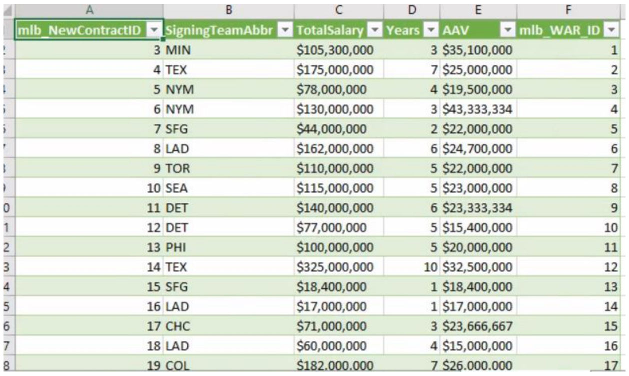
Name	Data	Schema	Item	Kind
mlb_NewContract	[Table]	dbo	mlb_NewContract	Table
mlb_Player	[Table]	dbo	mlb_Player	Table
mlb_PlayerPOS	[Table]	dbo	mlb_PlayerPOS	Table
mlb_QO	[Table]	dbo	mlb_QO	Table
mlb_Team	[Table]	dbo	mlb_Team	Table
mlb_WAR	[Table]	dbo	mlb_WAR	Table

Queries & Connections

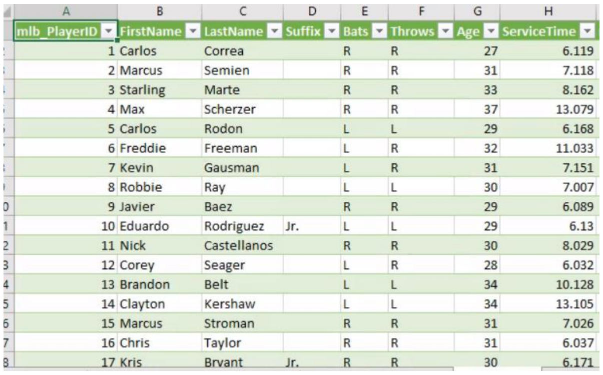
Queries | Connections

1 query

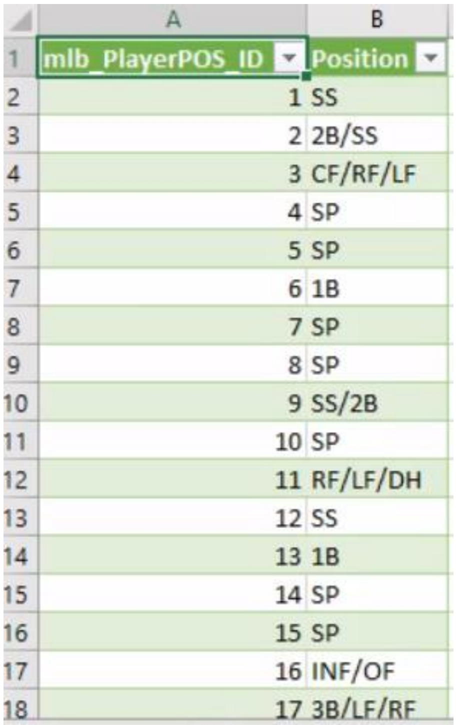
IST659_M408_gcconde
33 rows loaded.



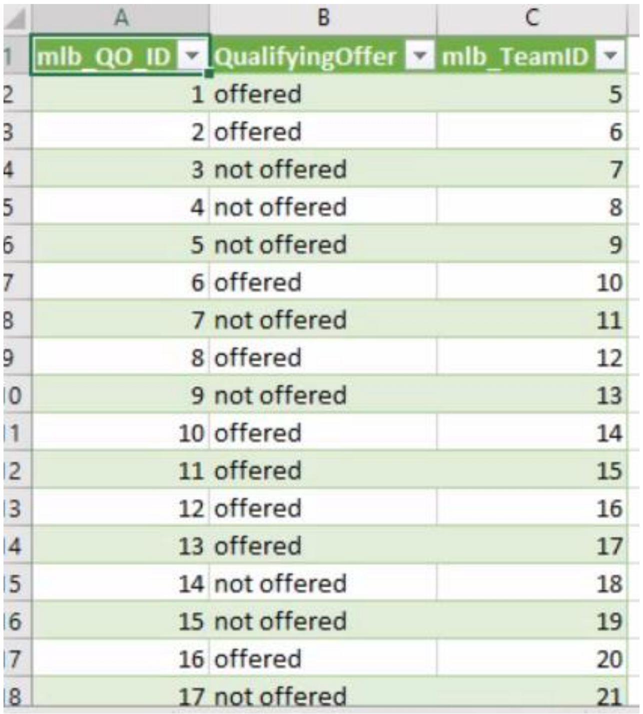
mlb_NewContractID	SigningTeamAbbr	TotalSalary	Years	AAV	mlb_WAR_ID
3	MIN	$105,300,000	3	$35,100,000	1
4	TEX	$175,000,000	7	$25,000,000	2
5	NYM	$78,000,000	4	$19,500,000	3
6	NYM	$130,000,000	3	$43,333,334	4
7	SFG	$44,000,000	2	$22,000,000	5
8	LAD	$162,000,000	6	$24,700,000	6
9	TOR	$110,000,000	5	$22,000,000	7
10	SEA	$115,000,000	5	$23,000,000	8
11	DET	$140,000,000	6	$23,333,334	9
12	DET	$77,000,000	5	$15,400,000	10
13	PHI	$100,000,000	5	$20,000,000	11
14	TEX	$325,000,000	10	$32,500,000	12
15	SFG	$18,400,000	1	$18,400,000	13
16	LAD	$17,000,000	1	$17,000,000	14
17	CHC	$71,000,000	3	$23,666,667	15
18	LAD	$60,000,000	4	$15,000,000	16
19	COL	$182,000,000	7	$26,000,000	17



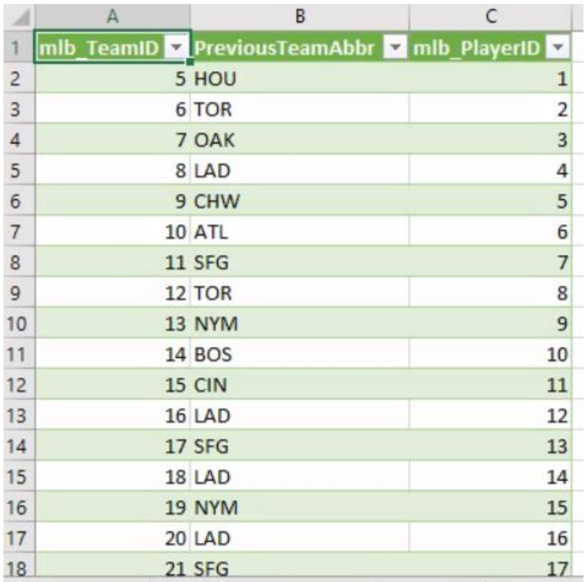
mlb_PlayerID	FirstName	LastName	Suffix	Bats	Throws	Age	ServiceTime
1	Carlos	Correa		R	R	27	6.119
2	Marcus	Semien		R	R	31	7.118
3	Starling	Marte		R	R	33	8.162
4	Max	Scherzer		R	R	37	13.079
5	Carlos	Rodon		L	L	29	6.168
6	Freddie	Freeman		L	R	32	11.033
7	Kevin	Gausman		L	R	31	7.151
8	Robbie	Ray		L	L	30	7.007
9	Javier	Baez		R	R	29	6.089
10	Eduardo	Rodriguez	Jr.	L	L	29	6.13
11	Nick	Castellanos		R	R	30	8.029
12	Corey	Seager		L	R	28	6.032
13	Brandon	Belt		L	L	34	10.128
14	Clayton	Kershaw		L	L	34	13.105
15	Marcus	Stroman		R	R	31	7.026
16	Chris	Taylor		R	R	31	6.037
17	Kris	Bryant	Jr.	R	R	30	6.171



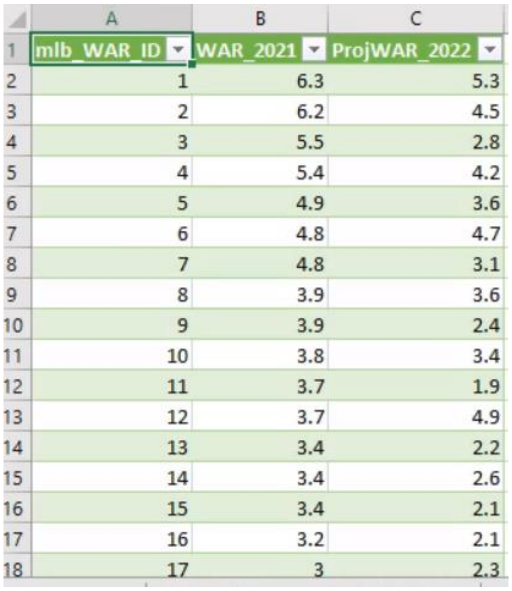
mlb_PlayerPOS_ID	Position
1	SS
2	2B/SS
3	CF/RF/LF
4	SP
5	SP
6	1B
7	SP
8	SP
9	SS/2B
10	SP
11	RF/LF/DH
12	SS
13	1B
14	SP
15	SP
16	INF/OF
17	3B/LF/RF



mlb_QO_ID	QualifyingOffer	mlb_TeamID
1	offered	5
2	offered	6
3	not offered	7
4	not offered	8
5	not offered	9
6	offered	10
7	not offered	11
8	offered	12
9	not offered	13
10	offered	14
11	offered	15
12	offered	16
13	offered	17
14	not offered	18
15	not offered	19
16	offered	20
17	not offered	21



1	mlb_TeamID	PreviousTeamAbbr	mlb_PlayerID
2	5	HOU	1
3	6	TOR	2
4	7	OAK	3
5	8	LAD	4
6	9	CHW	5
7	10	ATL	6
8	11	SFG	7
9	12	TOR	8
10	13	NYM	9
11	14	BOS	10
12	15	CIN	11
13	16	LAD	12
14	17	SFG	13
15	18	LAD	14
16	19	NYM	15
17	20	LAD	16
18	21	SFG	17



1	mlb_WAR_ID	WAR_2021	ProjWAR_2022
2	1	6.3	5.3
3	2	6.2	4.5
4	3	5.5	2.8
5	4	5.4	4.2
6	5	4.9	3.6
7	6	4.8	4.7
8	7	4.8	3.1
9	8	3.9	3.6
10	9	3.9	2.4
11	10	3.8	3.4
12	11	3.7	1.9
13	12	3.7	4.9
14	13	3.4	2.2
15	14	3.4	2.6
16	15	3.4	2.1
17	16	3.2	2.1
18	17	3	2.3

Reflection & Learning Goals

Throughout this project, some of the changes made were the entity names, the amount of data to be included, and data types within the entities. Changing the entity names 'Contract' to 'NewContract' relieved some confusion since the phrase 'Contract' was a function. Adding 'mlb_' to the beginning of entity names allowed created tables to be viewed in the Object Explorer together instead of scattered around tables.

The original data set contained 262 players, so minimizing it to the top 20 players based on 2021 WAR afforded less tediousness while providing adequate analysis and maintaining the integrity of the project. Altering fields that contained values in dollar figures from integers to varchars, such as QualifyingOffer, allowed for a smoother process since the QualifyingOffer fields provided the same figures for those who received them. This essentially made the data questions involving this field as yes or no questions.

Fields such as Suffix, Bats, and Throws were unnecessary to the scope of this project. Information regarding a player's injury history, such as instances where they were placed on the Injured List, the scope of these injuries, and games missed due to injury would be interesting to factor into this project. I believe that these factors contribute to a players future contract since teams would be weary of signing injury-prone players. Injuries also vary depending on a player's position, with pitchers tending to be the most injury prone. I believe that these factors could lead to more advanced data questions. Some examples include:

- Is a pitcher with an injury stint of over 60 days less likely to receive a Qualifying Offer than a position player with an injury stint of over 60 days?
- Are players over the age of 30 less likely to receive a long-term contract (5+ years) if they were placed on the Injured List two times or more?

The skills learned and utilized from this project include:

- Collection of data from databases that revolve around the industries of sports and finance.
- Utilizing relational database managing systems to create conceptual and logical models.
- The application of advanced SQL querying for physical database design, data creation, and data manipulation.
- The implementation of Microsoft Excel to assist in building a basic front end that provides a user interface for maintaining and reporting data.
- Communication of insights attained to stakeholders.

IST 707 – Applied Machine Learning

Course Description

This course presented a recapitulation of procedural machine learning techniques through specialized program packages, model building, and real-world applications. These technique established the foundation for data storytelling concepts, the discovery of trends using R and RStudio, and the ability to express technical findings to audiences.

Project Description

Goal: To determine the event of death by heart failure based on clinical measurements.

Contributors: Gianni Conde, Joshua Biggs-Bauer (jrbiggsb@syr.edu), Joaquin Rodarte (jrodarte@syr.edu)

About the Data: The data was retrieved from Kaggle and was originally collected from April to December of 2015 by the Allied Hospital in Faisalabad and the Faisalabad Institute of Cardiology, both residing in Punjab, Pakistan. The data contains patient medical information regarding heart failure with 299 patients (observations) and 12 clinical measurements (variables). These clinical measurements included age, anemia, creatinine phosphokinase (CPK), diabetes, ejection fraction, high blood pressure, platelets, serum creatinine, serum sodium, sex, smoking, time, and death event.

Age	Patient’s age (years)	**Sex**	female=0, male=1 (binary)
Anaemia	Decrease of hemoglobin (binary)	**Platelets**	Platelets in bloodstream (kilo platelets/mL)
High blood pressure	Hypertension (binary)	**Serum creatine**	Level of creatinine in bloodstream (mg/dL)
Creatinine phosphokinase (CPK)	Level of CPK enzyme in bloodstream (mcg/L)	**Serum sodium**	Level of sodium in bloodstream (mEq/L)

Diabetes	If a patient has diabetes (binary)	**Smoking**	If the patient smokes (binary)
Ejection fraction	% Of blood leaving the heart at each contraction	**Death Event**	If the patient died during the follow-up period (binary)

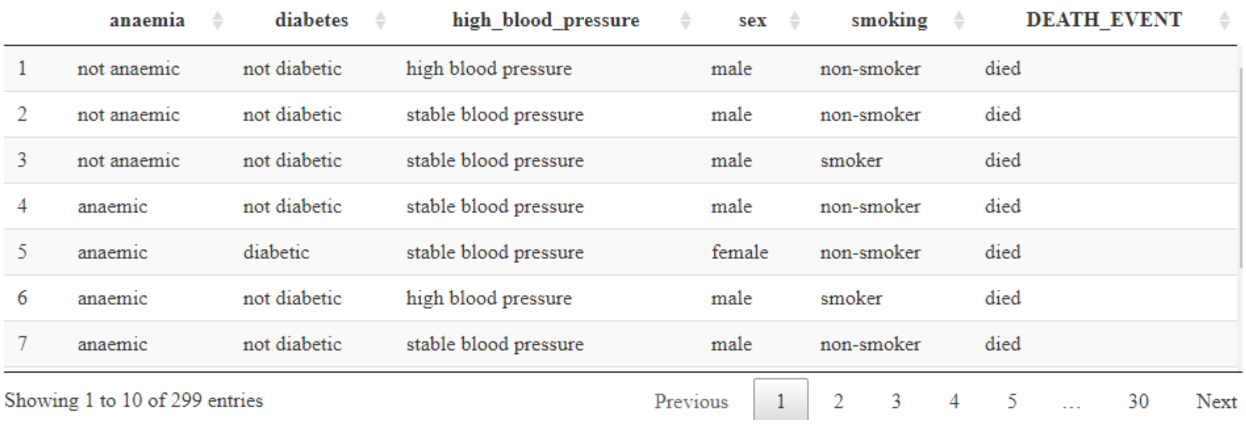
The table above depicts the names of these clinical measurements and a brief description. The binary variables in the data have values of 0 indicating no while 1 indicates yes. In the case of the binary sex variable, 0 indicates female while 1 indicates male.

Step 1: Data Cleansing and Preparation

Upon inspection, the data set contained no missing values. However, the time variable was difficult to interpret. Since this metric did not appear to be a defining characteristic of this analysis, the column was removed. Additionally, a patient ID index column beginning with 1 was included to solidify the data.

Since this analysis will contain models and methods pertaining to supervised and unsupervised machine learning, additional data preparations pertaining to each specific method will be necessary. For the supervised machine learning methods, the original data set was split into both training and testing data. The training data consisted of 80% of the overall data, while the testing data consisted of 20%.

The unsupervised machine learning methods include creating a new data frame consisting of only the binary clinical measurements and replacing the binary values of 0 and 1 with appropriate character descriptions.



	anaemia	diabetes	high_blood_pressure	sex	smoking	DEATH_EVENT
1	not anaemic	not diabetic	high blood pressure	male	non-smoker	died
2	not anaemic	not diabetic	stable blood pressure	male	non-smoker	died
3	not anaemic	not diabetic	stable blood pressure	male	smoker	died
4	anaemic	not diabetic	stable blood pressure	male	non-smoker	died
5	anaemic	diabetic	stable blood pressure	female	non-smoker	died
6	anaemic	not diabetic	high blood pressure	male	smoker	died
7	anaemic	not diabetic	stable blood pressure	male	non-smoker	died

Showing 1 to 10 of 299 entries

Step 2: Data Visualization and Exploration

Of the 13 clinical measurements, seven are considered to be nominal while six are binary. The nominal variables include id, age, creatinine phosphokinase (CPK), ejection fraction, platelets, serum creatinine, and serum sodium. The binary variables include anaemia, diabetes, high blood pressure, sex, smoking, and death event.

Two correlation plots were run to determine if there were any potential correlations between the data points. The first correlation plot (Figure 1.1) included all aspects of the data, while the second correlation plot (Figure 1.2) contained all aspects with the exclusion of time.

Figure 1.1

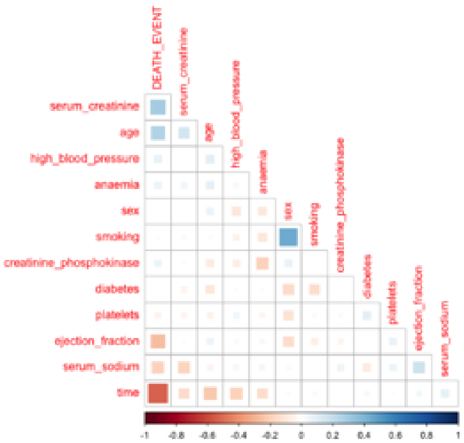


Figure 1.2

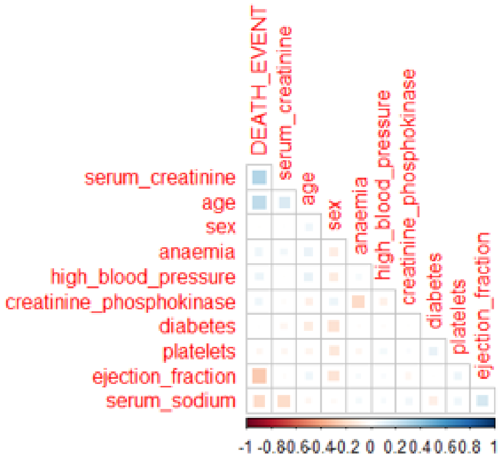
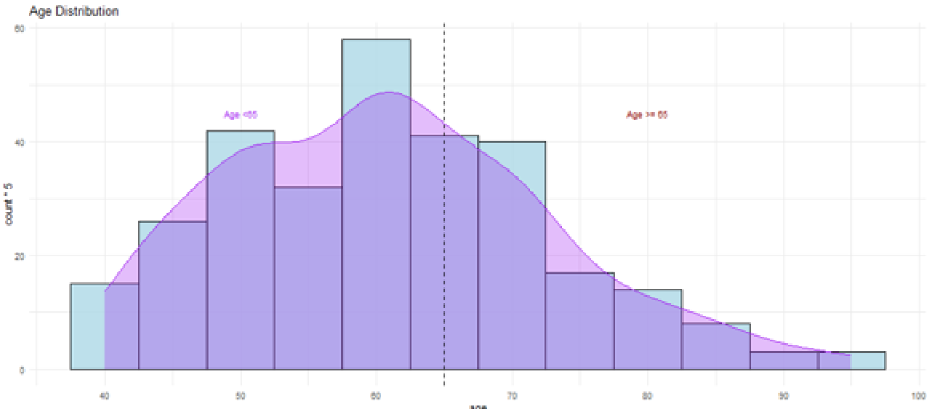


Figure 1.1 shows that smoking and sex had the highest positive correlation, followed by serum creatinine and death event, and age and death event. It also shows that time and death event are extremely negatively correlated. This implies that time could have miniscule relevance to the analyses, leading to its removal from the data frame. Figure 1.2 also shows that serum creatinine and death event are positively correlated, followed by age and death event, serum age and serum creatinine, and serum sodium and ejection fraction. The most negatively correlated data points in Figure 1.2 are ejection fraction and death event, implying that a patient's ejection fraction has little effect on their mortality rate.

Figure 1.3 was generated to graphically visualize the distribution of age and death event. The histogram shaded in blue displays the layer of patient age and the purple overlapping portion of the graph represents the distribution of death events among patients belonging to their age groups.

Figure 1.3



Since there were no participants under the age of 40 and over the age of 95, the overlapping distribution of death events among patient ages begin and end at those points.

Distributions of various clinical measurement against death even were visualized.

Figure 1.4

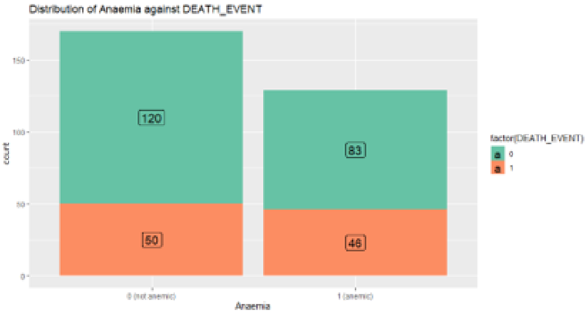


Figure 1.5

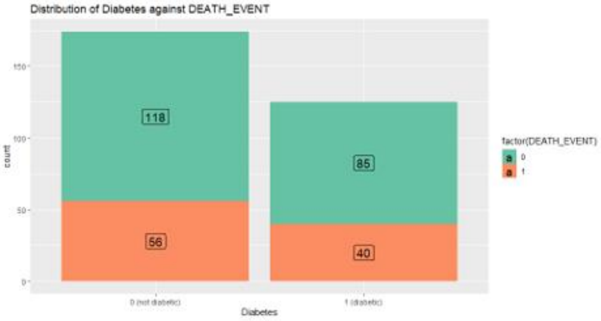


Figure 1.6

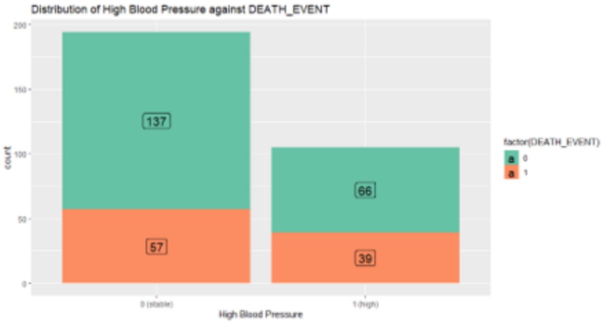


Figure 1.7

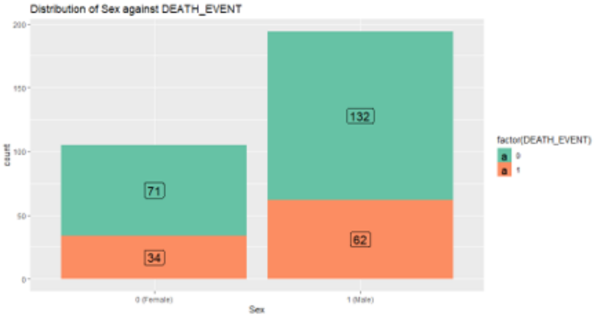
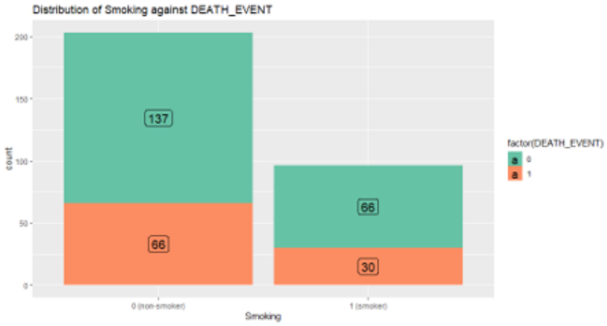


Figure 1.8



The distribution of anemia against death event (Figure 1.4) suggests that of the 299 patients, 170 were not anemic while 129 were anemic. Of the non-anemic patients, 120 survived and 50 died. Of the anemic patients, 83 survived and 46 died.

According to the distribution of diabetes against death event (Figure 1.5), 174 were non-diabetic and 125 were diabetic. Of the non-diabetic patients, 118 survived and 56 died. Of the patients who were diabetic, 85 survived while 40 died.

The distribution of high blood pressure against death event (Figure 1.6) suggests that 194 had stable blood pressure and 105 had high blood pressure. Of the patients with stable blood pressure, 137 survived and 57 died. Of the patients with high blood pressure, 66 survived while 39 died.

The distribution of sex against death event (Figure 1.7) states that 105 were female while 198 were male. In terms of the female patients, 71 survived while 34 died. In terms of the male patients, 132 survived while 62 died.

According to the distribution of smoking against death event (Figure 1.8), 203 were non-smokers while 96 were smokers. Regarding non-smokers, 137 survived while 66 died. Regarding those who were smokers, 66 survived while 30 died.

Next, the distribution of the data for patient age (Figure 1.9) revealed a range of 51 between the lowest and highest values present. The lowest age recorded is 40 years old and 91, the oldest age recorded and an outlier in the dataset.

Figure 1.9

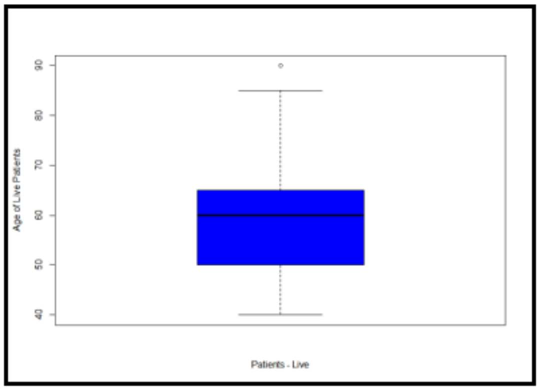
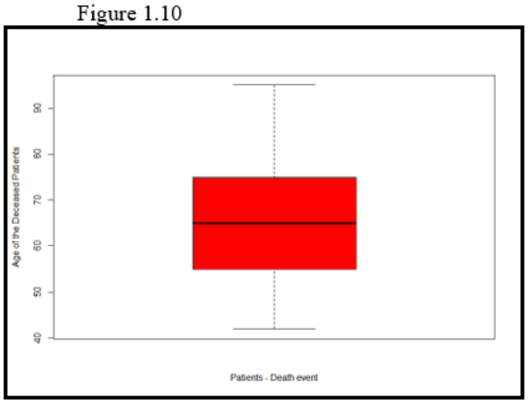


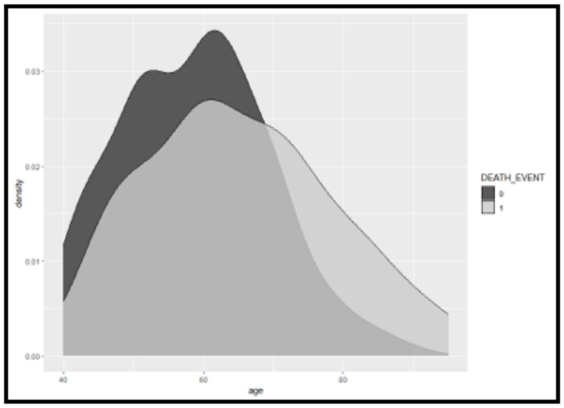
Figure 1.10



The patient death event data subset revealed that one of the outliers that reported a data event above the 90-year age. This is a clear sign that the distribution of the data ranges below 71 and 50 years of age with 40 being the youngest and 91 the oldest. The center of the data hovers around 60 years of age for the live patients (Figure 1.10). The patient live data subset revealed that an outlier that reported no data event was of the age of 91. This is a clear sign that the distribution of the data ranges below 71 and 50 years of age with 40 being the youngest and 91 the oldest. The center of this boxplot represents a higher mean for the death event subset around 5 years difference.

Given that age has tendency to report younger patients as less commonly to die from heart failure than older patients (Figure 1.11), the decision tree model analysis may be hampered by the data category as to skew results away from specific medical data. This may include substance levels in the body, lifestyle, or underlying conditions in the data set which ultimately are best suited to medical treatment.

Figure 1.11



Step 3: Random Forest

The initial random forest model (Figure 2.1), containing a seed set at 9, displayed a consensus of the results of various samples from a group of many decision trees. The confusion matrix (Figure 2.2) results stated that the random forest model had a sensitivity of 97.7%, specificity of 52.6%, precision of 82.4%, Kappa of 0.57, and an accuracy of 83.9%. Thus, the model could predict a death event of about 84% of the time.

Figure 2.1

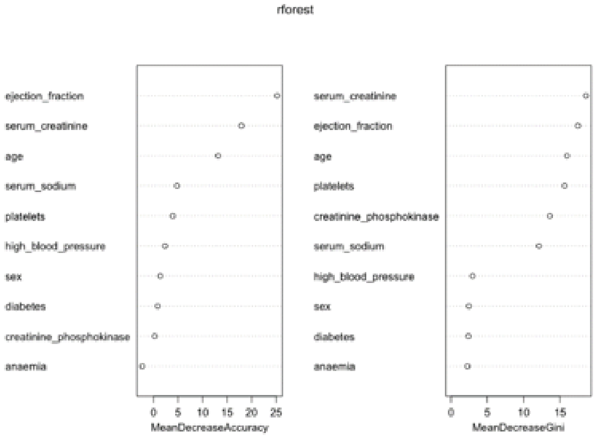
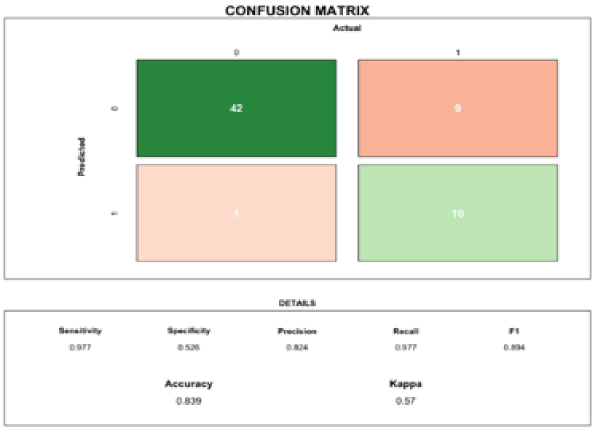


Figure 2.2



Sensitivity	Specificity	Precision	Recall	F1
0.977	0.526	0.824	0.977	0.894

Accuracy	Kappa
0.839	0.57

The second random forest model (Figure 2.3) contained a seed set at 20 and a data split where the training data held 70% of the data, while the testing data held 30%. The confusion matrix (Figure 2.4) results stated that the random forest model had a sensitivity of 98.2%, specificity of 35.7%, precision of 75%, Kappa of 0.398, and an accuracy of 77.1%. Thus, the model could predict a death event of about 77% of the time.

Figure 2.3

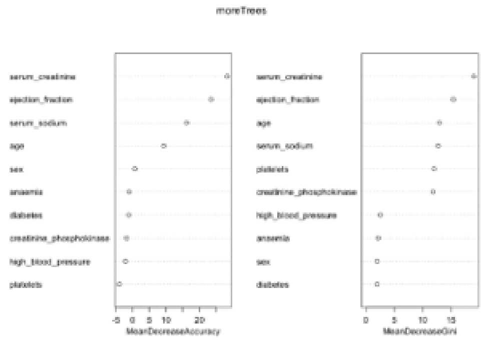
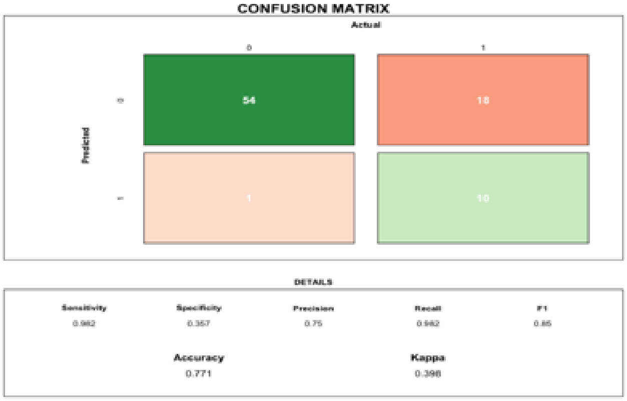


Figure 2.4



Despite the second model's adequacy, it appears that the initial random forest yielded the best model.

Step 4: Support Vector Machine (SVM)

With the first SVM model (Figure 2.5), the seed was set at nine with a data split where the training data consisted of 80% of the data and 20% assigned to the testing data. The model could predict a death event 78 percent of the time.

The second SVM model (Figure 2.6), had a seed set at twenty with a data split where the training data consisted of 70% of the data and 30% assigned to the testing data. The model could predict a death event 77% of the time.

Figure 2.5

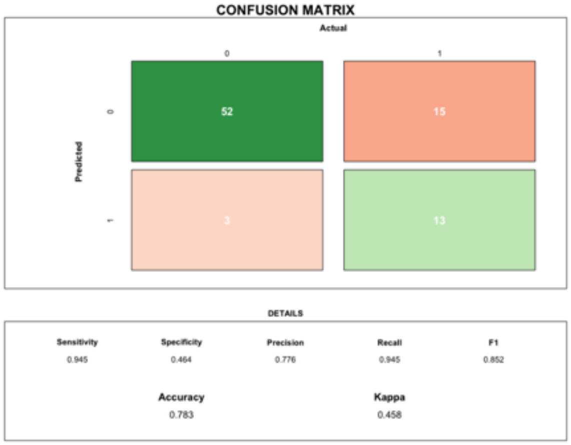
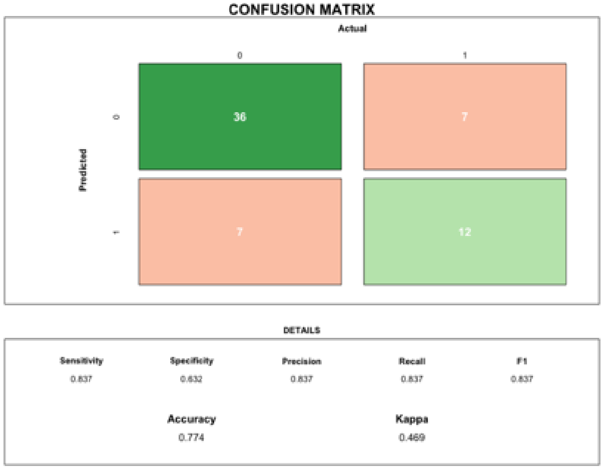


Figure 2.6

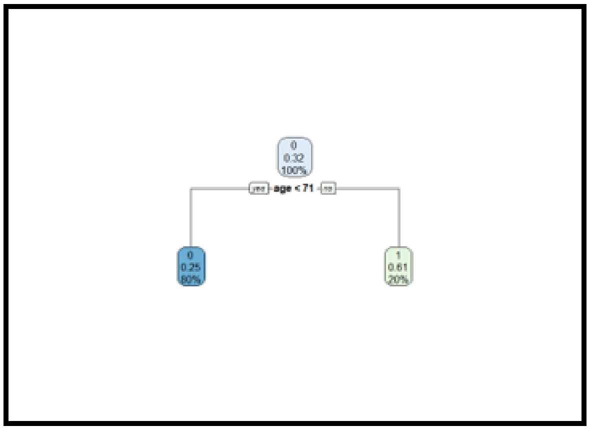


It appears that the SVM models did not perform at the same levels as the random forest models.

Step 5: Decision Tree

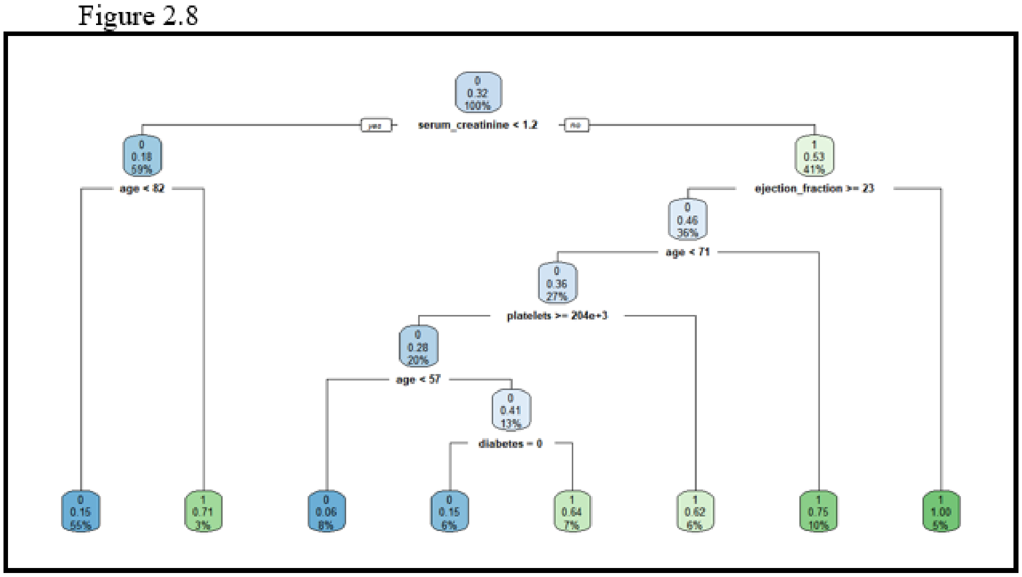
Several factors needed to be considered when creating a decision tree analysis model. First, there needed to be a way to remove age as a factor considering it does not provide any insight into a patient's overall health alone. The first decision tree (Figure 2.7) examines this occurring in the data, for any patients over the age of 71, there exists a higher likelihood of a death event in the dataset.

Figure 2.7



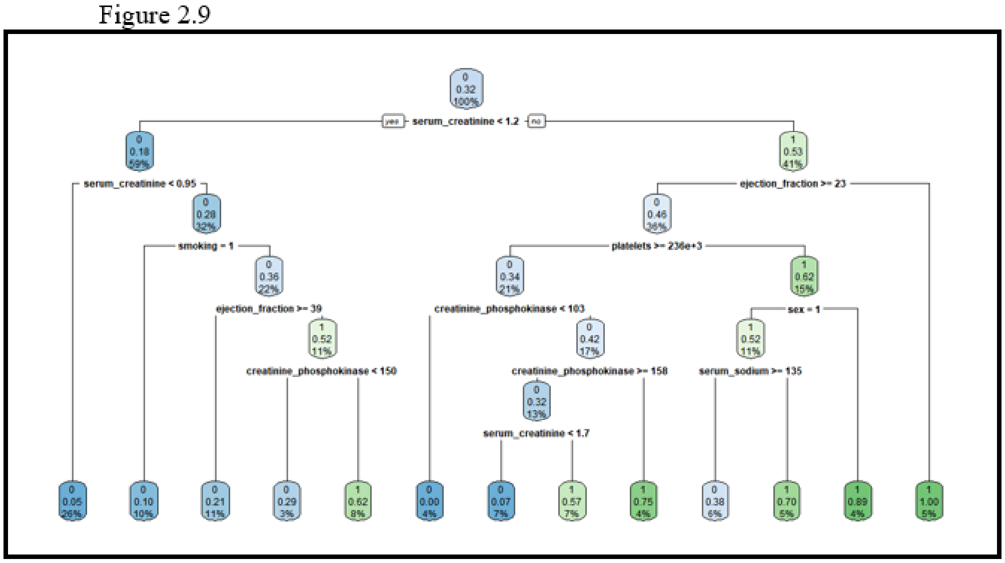
For the decision tree analysis, it was crucial to remove certain data columns as they had no relevance to the data or outcome. This included the time and patient ID values. In this model (Figure 2.8), it is revealed age does contribute as a significant node to determine death event, and it is not preferable to have a non-medical related data category dominating the data analysis model. Information that is actionable, informative and can be treated or examined is preferred. This model reported a 61% accuracy.

Figure 2.8



For the decision tree analysis, it was critical to remove non-informative data columns as they had no relevance to the focus of this analysis. This the second data subset excluded the time, patient ID, and age column values. This model (Figure 2.9) reported a 74% accuracy.

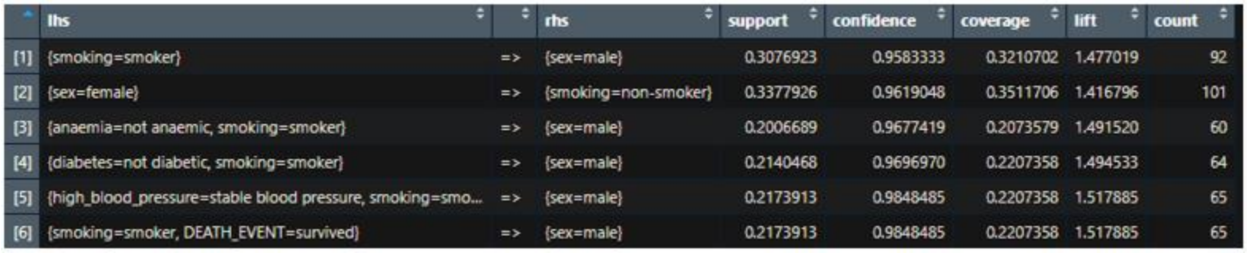
Figure 2.9



Step 6: Association Rule Mining

Association rule mining was performed on the binary clinical measurements of the heart failure data set, including anemia, diabetes, high blood pressure, sex, smoking, and death event. After various adjustments of the Apriori algorithm's parameters, a successful set of 20 rules were generated after setting the support to 0.13 and confidence to 0.9.

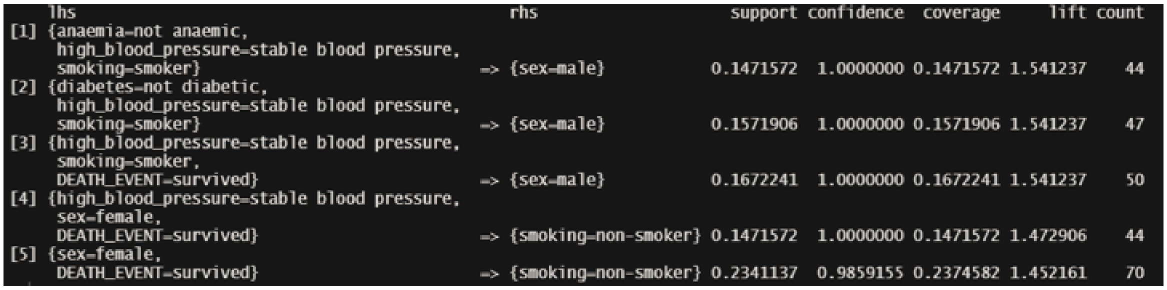
Table 2.1



	lhs		rhs	support	confidence	coverage	lift	count
[1]	{smoking=smoker}	=>	{sex=male}	0.3076923	0.9583333	0.3210702	1.477019	92
[2]	{sex=female}	=>	{smoking=non-smoker}	0.3377926	0.9619048	0.3511706	1.416796	101
[3]	{anaemia=not anaemic, smoking=smoker}	=>	{sex=male}	0.2006689	0.9677419	0.2073579	1.491520	60
[4]	{diabetes=not diabetic, smoking=smoker}	=>	{sex=male}	0.2140468	0.9696970	0.2207358	1.494533	64
[5]	{high_blood_pressure=stable blood pressure, smoking=smo...	=>	{sex=male}	0.2173913	0.9848485	0.2207358	1.517885	65
[6]	{smoking=smoker, DEATH_EVENT=survived}	=>	{sex=male}	0.2173913	0.9848485	0.2207358	1.517885	65

The top five rules in terms of confidence (Table 2.2) displayed various rules. For example, the 4th rule states that if a patient has stable blood pressure, is female, and survived, they are 100% likely to be a non-smoker.

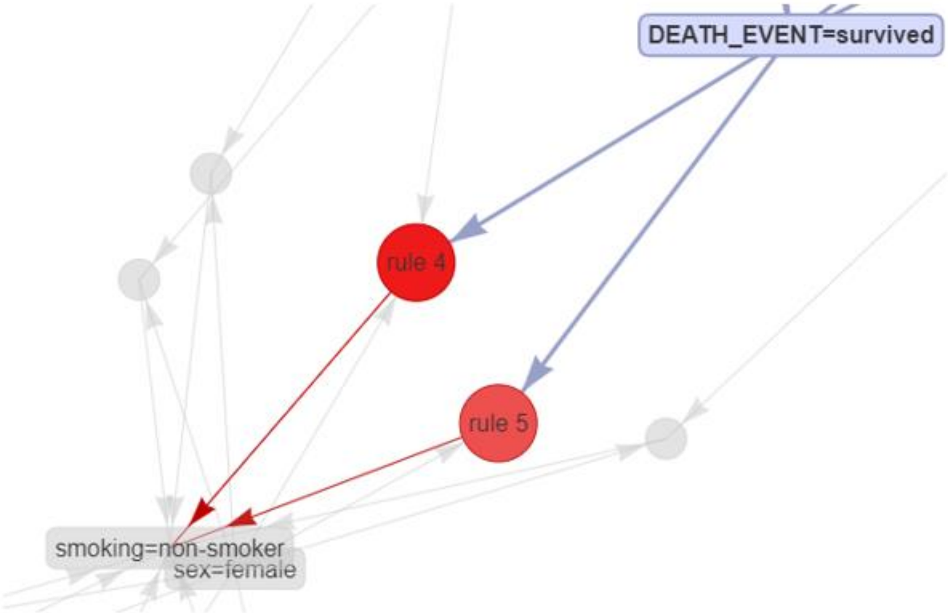
Table 2.2



```
    lhs                                          rhs                    support confidence  coverage     lift count
[1] {anaemia=not anaemic,
     high_blood_pressure=stable blood pressure,
     smoking=smoker}                          => {sex=male}           0.1471572  1.0000000 0.1471572 1.541237    44
[2] {diabetes=not diabetic,
     high_blood_pressure=stable blood pressure,
     smoking=smoker}                          => {sex=male}           0.1571906  1.0000000 0.1571906 1.541237    47
[3] {high_blood_pressure=stable blood pressure,
     smoking=smoker,
     DEATH_EVENT=survived}                    => {sex=male}           0.1672241  1.0000000 0.1672241 1.541237    50
[4] {high_blood_pressure=stable blood pressure,
     sex=female,
     DEATH_EVENT=survived}                    => {smoking=non-smoker} 0.1471572  1.0000000 0.1471572 1.472906    44
[5] {sex=female,
     DEATH_EVENT=survived}                    => {smoking=non-smoker} 0.2341137  0.9859155 0.2374582 1.452161    70
```

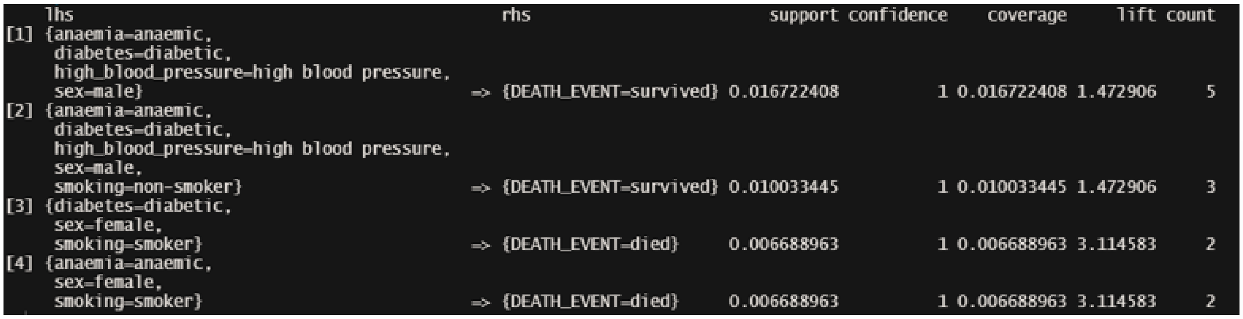
An HTML widget (Figure 2.10) of the top 5 rules was generated. It appears that death event, indicating survival in this scenario, is pointing towards a dark red circle, which represents a strong rule 4 stated previously. Rule 4 points towards patients being non-smokers, implying that being a non-smoker can lead to patient survival.

Figure 2.10



Next, the RHS of the Apriori algorithm was set to death event to discover rules that associated clinical variables to patient mortality. For example, the third rule states that if a patient is diabetic, female, and is a smoker, she is 100% likely to have died.

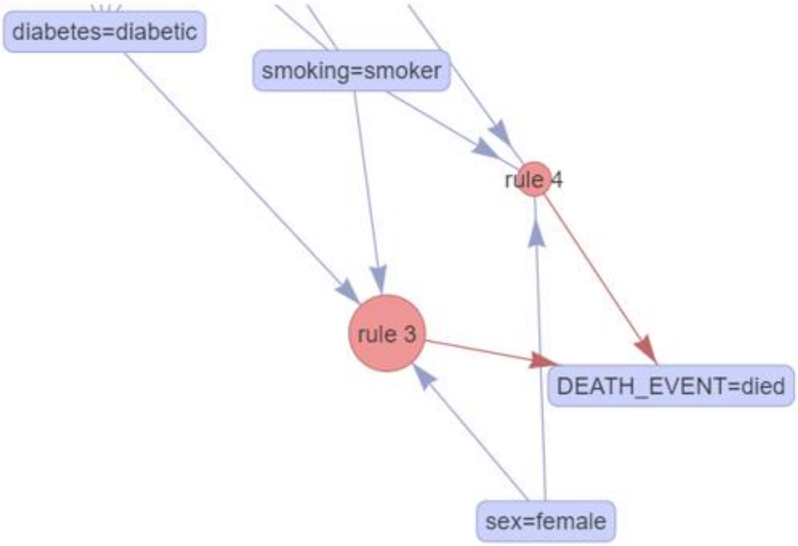
Table 2.3



```
    lhs                                            rhs                        support confidence    coverage     lift count
[1] {anaemia=anaemic,                                                                                                       
     diabetes=diabetic,                                                                                                     
     high_blood_pressure=high blood pressure,                                                                               
     sex=male}                                  => {DEATH_EVENT=survived} 0.016722408          1 0.016722408 1.472906     5
[2] {anaemia=anaemic,                                                                                                       
     diabetes=diabetic,                                                                                                     
     high_blood_pressure=high blood pressure,                                                                               
     sex=male,                                                                                                              
     smoking=non-smoker}                        => {DEATH_EVENT=survived} 0.010033445          1 0.010033445 1.472906     3
[3] {diabetes=diabetic,                                                                                                     
     sex=female,                                                                                                            
     smoking=smoker}                            => {DEATH_EVENT=died}     0.006688963          1 0.006688963 3.114583     2
[4] {anaemia=anaemic,                                                                                                       
     sex=female,                                                                                                            
     smoking=smoker}                            => {DEATH_EVENT=died}     0.006688963          1 0.006688963 3.114583     2
```

Another HTML widget was created for when the RHS (Figure 2.11) had been set to death event. It appears that diabetic, smoker, and female are pointing towards Rule 3, which then points to death event = died.

Figure 2.11



Reflection & Learning Goals

There appears to be a connection between serum creatinine and ejection fraction abnormalities and a higher chance of heart failure and death. While this is the case, other intriguing factors did not impact a possible death event as highly. Diabetes and high blood pressure did not seem to contribute to heart failure and death as the variables mentioned previously. While these are by no means something that can be ignored, it is intriguing that they did not play as big of a role.

Initial analysis using the decision tree model revealed that age is the most dominant factor considering heart failure death events for patients suffering from cardiovascular ailments. Within this category of medical conditions, it is understandable that older patients are more likely to die from heart failure. In the heart failure dataset, ages ranged from 40 to 91 years old. A distribution of the ages between dead and surviving patients varied with an average difference of 5 years, surviving patients leaning toward the younger side. Ruling out age as a factor helped better accommodate the analysis to more relevant medical test and prognosis with variables such as ejection fraction and serum creatinine being a better variable to monitor and target medically. Given serum creatinine Level of creatinine in the bloodstream above of 1.2 mg/dL and Ejection fraction value above 23, a patient would follow a branch that leads to a death event. This is followed by the nodes of a smoker and a significant level of platelets (2.3M) in the patient's blood.

In terms of the binary clinical measures, there appears to be a prevalent connections between smoking and death events resulting in patient death. There were various instances of strong associations between patient survival when the heart failure patient in question is a non-smoker.

The skills learned and utilized from this project include:

- Collection, storage, and accessing medical data.
- The application of advanced R packages such as ggplot2, arules, arulesViz, DescTools, corrplot, skimr, patchwork, randomForest, RColorBrewer, and dplyr,

- Visualization of various data frames and distributions to attain valuable and intuitive insights.
- Supervised and unsupervised machine learning methods such as Random Forest, SVM, Decision Trees, and Association Rule Mining.
- The strengths and weaknesses of supervised and unsupervised machine learning.
- Interpretation of analytical results regarding the medical field.

IST 736 – Text Mining

Course Description

This course introduced the fundamental ideas and processes for analyzing large amounts of text data through the use of clustering, document representation and extraction, text classification, and topic modeling. These techniques were applied alongside various visualization tools in Python to uncover interesting occurrences in real-world applications.

Project Description

Goal: To determine movie recommendations based on movie genre and description using supervised and unsupervised machine learning techniques.

Contributors: Gianni Conde, Sana Khan (skhan53@syr.edu), Sahil Nanavaty (sknanava@syr.edu)

About the Data: The data was retrieved from Kaggle and originated from IMDB.com. The original corpus contained 16 separate CSV files that each represented a genre. These genres included action, adventure, animation, biography, crime, family, fantasy, film-noir, history, horror, mystery, romance, sci-fi, sports, thriller, and war. Each file contained varying amounts of rows and 14 columns. The number of rows between each file ranged from 987 to 52,618. The columns represented movie ID, movie name, year, certificate, runtime, genre, rating, description, director, director ID, star, star ID, votes, and gross in USD.

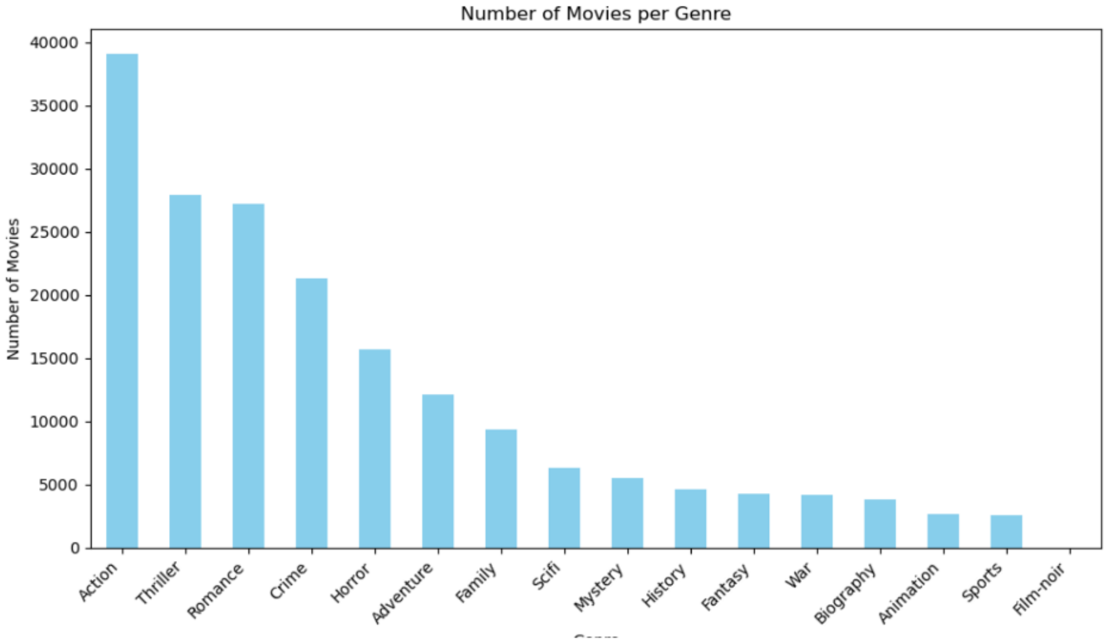
Step 1: Data Cleansing and Preparation

Upon iteration, it appeared that many movies in each file contained multiple genres in their genre column. For simplicity, the genres were changed to match their respective files. Each file was then concatenated into a single data frame, resulting in it having 368,300 rows and 14 columns. Unwanted values such as special characters, non-numerical characters in numerical categories, unneeded IDs, missing or duplicate values, time metrics, and oddly named column names were either removed, replaced, or altered. This resulted in a data frame with a shape of 142,626 rows and 11 columns.

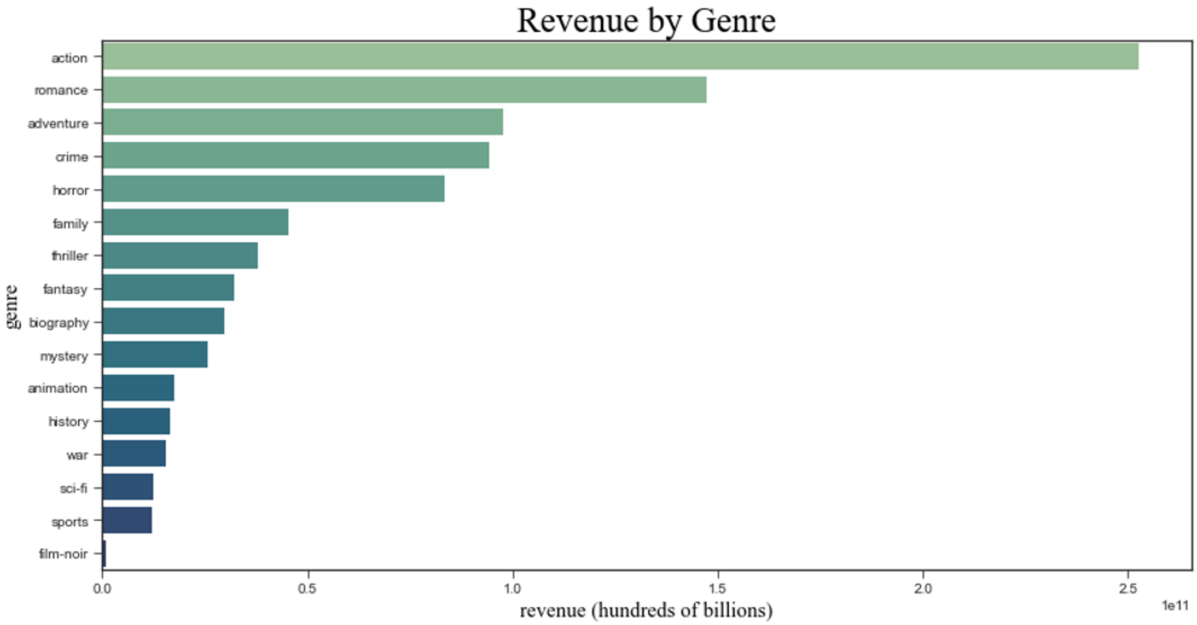
Since the goal of the analysis is to predict genre based on movie description, all other columns were removed. The genres of interest were crime, horror, romance, sci-fi, and sports, so all others were removed. The description field was then vectorized using Count Vectorizer and assigned to a new data frame. Stemming and removing numbers and stop words were then removed. The vectorized data was sampled to only contain 30% of the original data.

Step 2: Data Visualization and Exploration

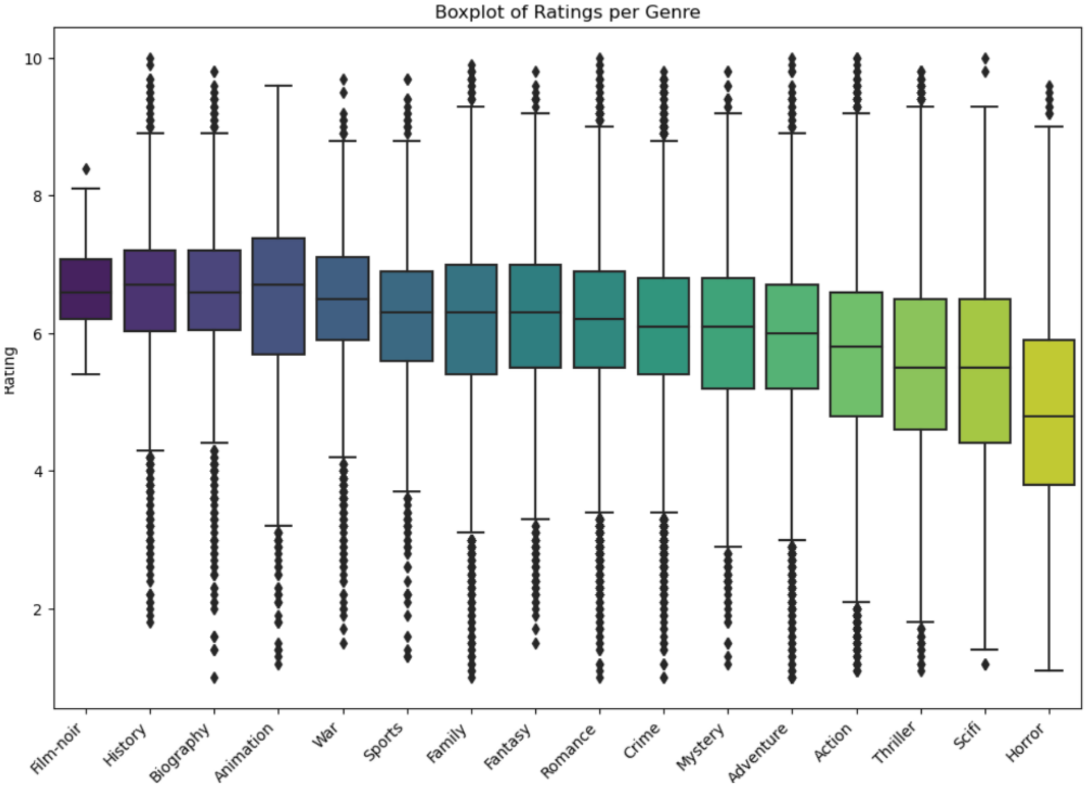
Prior to the removal of columns, exploratory analyses were conducted.



The five highest counts of movies were action, thriller, romance, crime, and horror. The lowest were film-noir, sports, animation, biography, and war.



Action movies also have the highest revenue of all the genres, followed by romance, adventure, crime, and horror. Film-noir, sports, sci-fi, war, and history generated the least revenue.



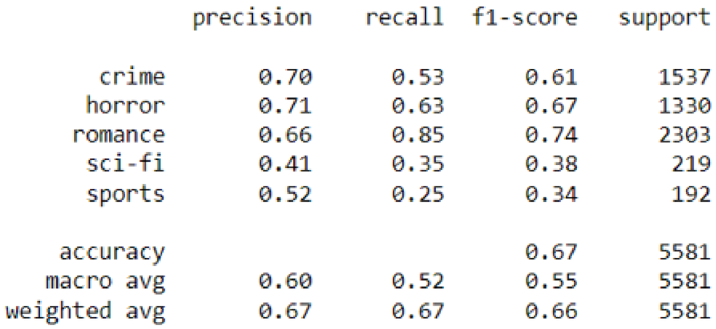
Film-noir had the highest average rating while all other genres had a relatively larger spread of ratings. Horror movies had the lowest average ratings across all genres.

Step 3: The data frame was split into testing and training sets.

Step 4: Multinomial Naïve Bayes

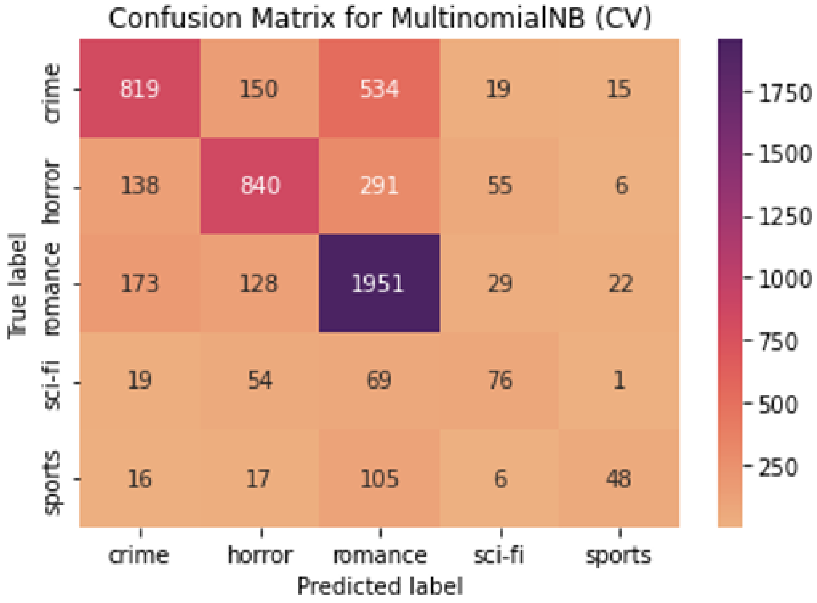
After instantiating and fitting the multinomial Naïve Bayes classifier with CountVectorizer to the vectorized data, a prediction was formed from the unlabeled test data based on log probabilities. This yielded an accuracy score of approximately 66.91%, which fell just below the industry standard (70-90%). A classification report was compiled to analyze the model's precision, recall, F1-scores, and support.

Figure: Classification Report for Multinomial Naïve Bayes



```
              precision    recall  f1-score   support

       crime       0.70      0.53      0.61      1537
      horror       0.71      0.63      0.67      1330
     romance       0.66      0.85      0.74      2303
      sci-fi       0.41      0.35      0.38       219
      sports       0.52      0.25      0.34       192

    accuracy                           0.67      5581
   macro avg       0.60      0.52      0.55      5581
weighted avg       0.67      0.67      0.66      5581
```

The classification report showed that all three scores were about the same (60-67%) in terms of weighted average. The precision, recall, and F1-score values for each genre yielded higher scores for those with larger support compared to those with smaller support. In terms of precision, crime and horror scored the highest, followed closely by romance. Regarding recall, romance had a score of 85%, well above the industry standard (60-80%). Romance also secured the top scorer (74%) in terms of F1-score, with anything about 70% being deemed as good. The two genres that essentially weighed down the averages were sci-fi and sports with scores ranging from 25-52% across all percentage-based metrics. This could be attributed to their significantly smaller support compared to the other genres.

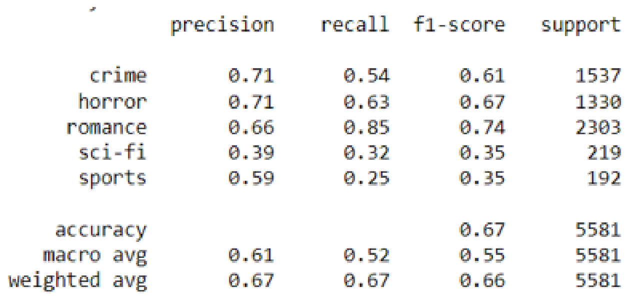


The matrix shows that of the 5,581 descriptions in the testing data, 3734 were predicted correctly. 819 crime descriptions were predicted correctly while 718 were incorrectly predicted. 840 horror descriptions were predicted correctly while 490 were incorrectly predicted. 1951 romance descriptions were predicted correctly while only 352 were incorrect. 76 sci-fi descriptions were predicted correctly compared to 143 incorrect predictions. Lastly, 48 sports descriptions were predicted correctly compared to 144 being incorrect.

Step 5: Bernoulli Naïve Bayes

After instantiating and fitting the Bernoulli Naïve Bayes classifier with CountVectorizer to the vectorized data frame, a prediction was formed from the unlabeled test data based on log probabilities. This yielded an accuracy score of approximately 66.87%, which fell just below the industry standard (70-90%) and was almost identical to the multinomial Naive Bayes model.

Figure: Classification Report for Bernoulli Naïve Bayes

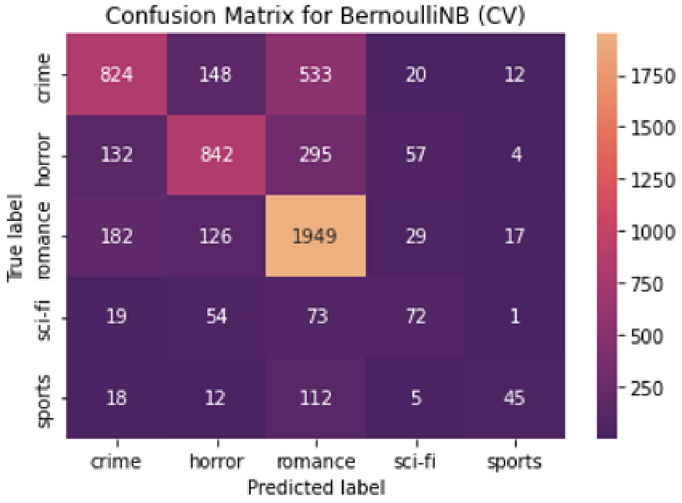


```
              precision    recall  f1-score   support

       crime       0.71      0.54      0.61      1537
      horror       0.71      0.63      0.67      1330
     romance       0.66      0.85      0.74      2303
      sci-fi       0.39      0.32      0.35       219
      sports       0.59      0.25      0.35       192

    accuracy                           0.67      5581
   macro avg       0.61      0.52      0.55      5581
weighted avg       0.67      0.67      0.66      5581
```

The report shows that all three scores were about the same (60-67%) in terms of weighted average. Like the previous model, precision, recall, and F1-score values for each genre yielded higher scores for those with larger support than those with smaller support. While the results were nearly the same as the previous model across the board, the differences were this model's precision scores for the sci-fi and sports genre. Sci-fi had a precision score of 39%, which was a 2% decrease from the previous model, and sports had a precision score of 59%, which was a 7% increase compared to the previous model.

In terms of precision, crime and horror scored the highest, followed closely by romance. Regarding recall, romance had a score of 85%, well above the industry standard (60-80%). Romance also secured the top scorer (74%) in terms of F1-score, with anything about 70% being deemed as good. The two genres that weighed down the averages were sci-fi and sports with scores ranging from 25-59% across all percentage-based metrics. This could be attributed to their significantly smaller support compared to the other genres.

Figure. Confusion Matrix for Bernoulli Naïve Bayes



The matrix shows that of the 5,581 descriptions in the testing data, 3732 were predicted correctly. 824 crime descriptions were predicted correctly while 713 were incorrectly predicted. 842 horror descriptions were predicted correctly while 488 were incorrectly predicted. 1949 romance descriptions were predicted correctly while only 354 were incorrect. 72 sci-fi descriptions were predicted correctly compared to 147 incorrect predictions. Lastly, 45 sports descriptions were predicted correctly compared to 147 being incorrect.

When comparing the results of both Naive Bayes models, it appears that they are both nearly identical. Their similar accuracy score implies that the features are not highly correlated and that the genres were well separated by movie description features. The low classification report scores for sci-fi and sports were most likely due to low support.

Step 6: Support Vector Machine (SVM)

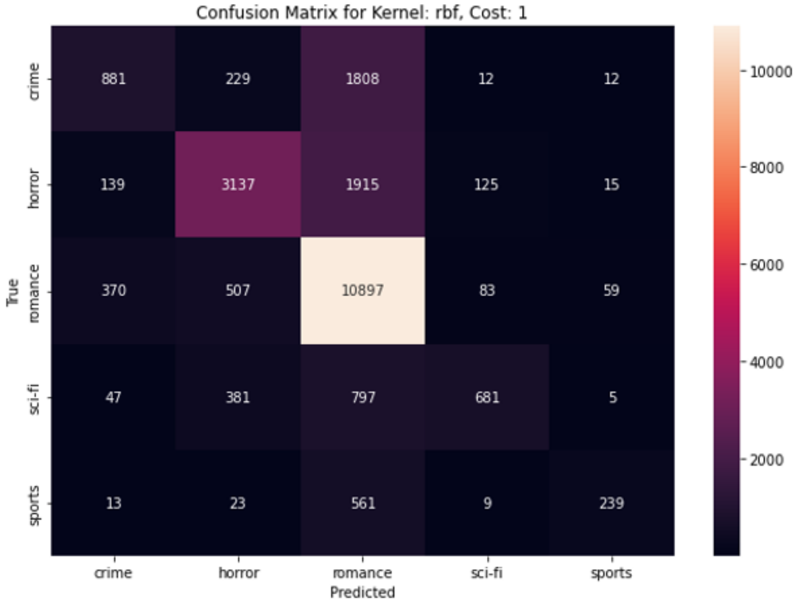
The SVM classifier was trained on the feature vectors derived from the movie descriptions, with genre labels serving as the target variable. Throughout this stage, experimentation with three kernels – linear, radial basis function (RBF), and polynomial – alongside various costs were conducted to determine the optimal approach for the dataset. The linear kernel, while simplistic, was expected to offer the most computationally efficient model. In contrast, the RBF and polynomial kernels were hypothesized to yield higher accuracy scores after precise parameter tuning.

Based on the results, the effectiveness of various SVM model configurations can be assessed at a high level. Generally, increasing the cost parameter for the linear kernel from 0.1 to 10 led to a slight improvement in accuracy. This is because higher values of the cost parameter within the linear kernel allow the model to penalize misclassified points more heavily, leading to a tighter decision boundary (and potentially better generalization). However, the recorded improvements in accuracy were relatively small across different cost values. This suggests that the linear kernel may have reached its performance limit on this dataset.

Regarding the RBF kernel, the accuracy score improved notably when transitioning from a low-cost value of 0.1 to a higher cost value of 1. This indicates that a moderate cost value was potentially more suitable for this kernel on this dataset. In other words, a stricter margin (achieved with a higher cost parameter) was more effective in capturing relationships between features and labels. Despite this observation, increasing the cost parameter further to 10 resulted in a decrease in accuracy. This could indicate that the decision boundary became too rigid, leading to overfitting on the training data and reduced generalization.

Finally, the polynomial kernel displayed a generally lower accuracy compared to that of the linear and RBF kernels across all cost values. This could be attributed to the nature of the dataset as well as the specific parameters used. Increasing the cost value from 0.1 to 1 led to a significant improvement in accuracy, suggesting that a moderate cost value allowed was more appropriate for the polynomial kernel. Ultimately, increasing the cost parameter to 10 resulted in a decrease in accuracy, similar to the behavior observed with the RBF kernel. This indicates that the decision boundary may have been too complex which led to overfitting.

Figure: Confusion matrix for SVM (highest performing configuration)



Step 7: K-Means Clustering

Since K-Means clustering is an unsupervised method, the testing set's labels were dropped. The elbow method was used to determine the optimal number of clusters for the model. It first assigns several clusters and the sum of squared distances of the data points to their nearest cluster center. Once the values were plotted, the optimal number of clusters were found at the elbow point from the curve on the graph. The bend in the curve represents the point where increasing the number of clusters no longer provides significant improvement towards fitting the model. Therefore, the optional number of clusters is either three or four.

Figure: Elbow method plot (K=3)

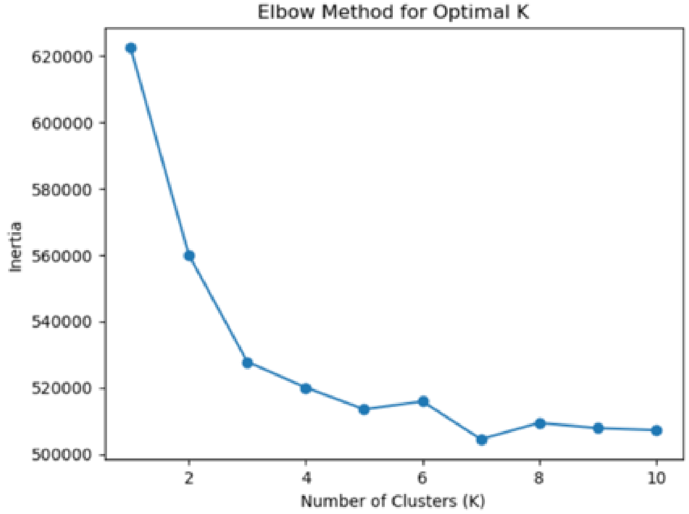


Figure: Clustering plot (K=3)

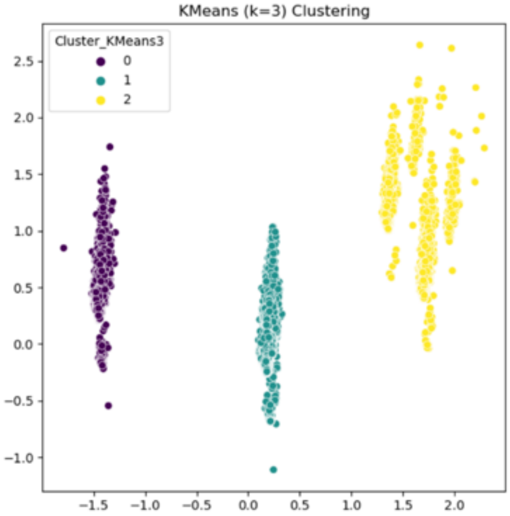
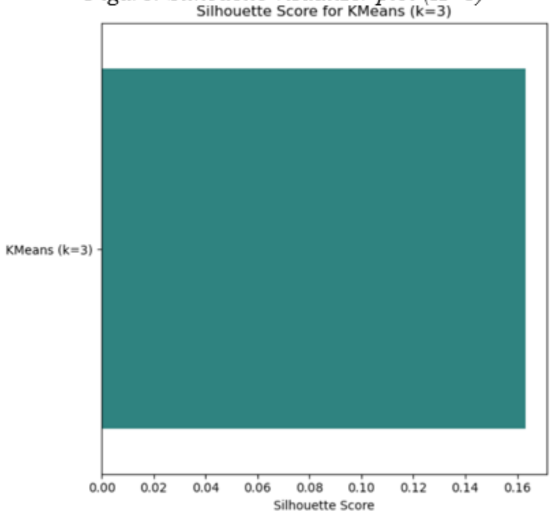


Figure: Silhouette visualizer plot (K=3)



Based on the plot, we can see that cluster 0 and 1 have a lot of similarities with a few outliers. However, cluster 2 is spread out and does not share as many similarities among the movie genres. If a viewer watched a movie from 0 and wanted to watch another movie from this cluster, the recommendation would surely be a good match. This also applies to cluster 1. Although, given the disparity among the last cluster, it may be difficult to provide an accurate recommendation for the viewer. The silhouette score of 0.16 further suggests that while the clusters are distinct enough to be separate, the separation is not particularly strong. This means that the descriptions contain either ambiguous terms or the genres are not distinctly different in the features used for clustering.

Figure: Clustering plot (K=4)

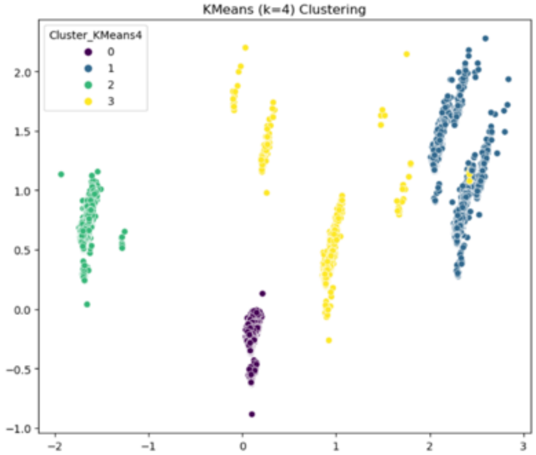
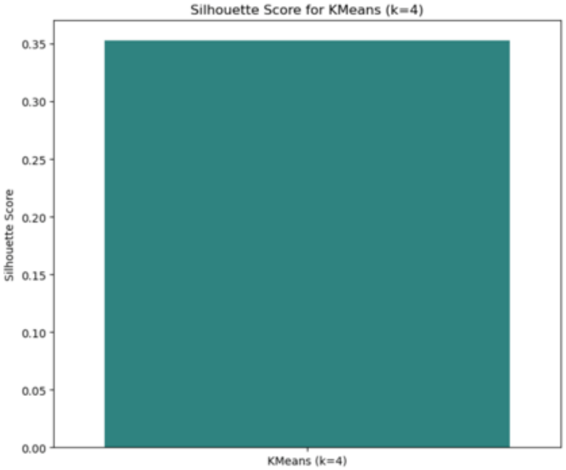


Figure: Silhouette visualizer plot (K=4)

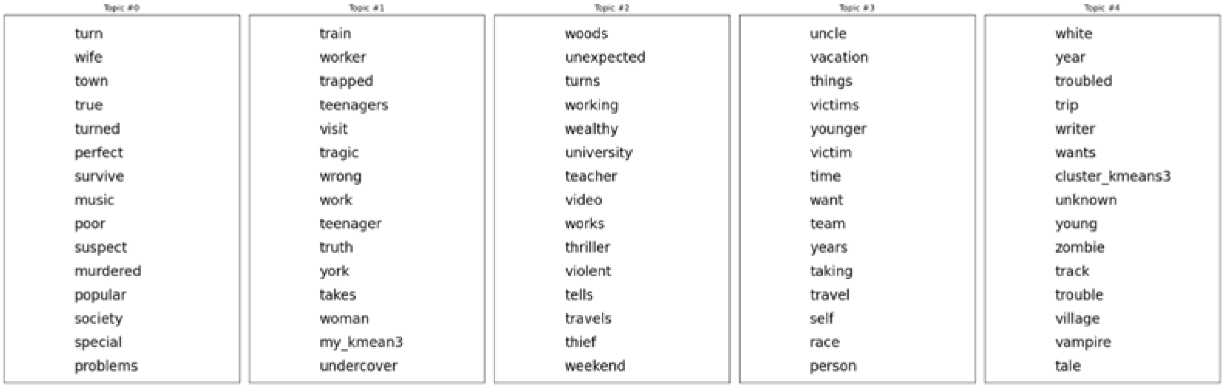


The model was also run with 4 clusters as the elbow appears to plateau between 3 and 4. There are improvements in the clusters due to the addition of a new cluster absorbing some of the data points that were found in other clusters. Cluster 1 does have some overlap with cluster 3 but is otherwise more defined in comparison to the previous plot. Cluster 3 has a relatively wider spread, indicating there may not be as many similarities amongst them. Clusters 0 and 2 contain some outliers, but otherwise have a more defined shape which indicates some similarities in the words found in the descriptions. The silhouette for 4 clusters provided an improved score of 0.35. This score suggests that the clusters are reasonably well separated and that the points within each cluster are relatively close to each other.

Step 8: Latent Dirichlet Allocation (LDA)

The LDA model suggests that it is difficult to identify a clear genre for either topic. Topic 0 could be crime films. Words like *suspect*, *murdered* and *survive* would align best with movies containing criminal elements. Topic 2 could be related to horror movies. Words such as *thriller*, *violent*, *woods,* and *unexpected* indicate something fearful which is the basis of the horror genre. Topic 3 could potentially be about sports movies, with words like *team*, *race*, and *time*. Topic 4 seems best aligned to sci-fi with *zombie vampire,* and *tale*. Topic 1 does not appear to have a clear genre. However, based on words like *woman*, *tragic*, and *teenager,* the topic could be pertaining to romance movies.

Figure: Topic groupings



Topic #0	Topic #1	Topic #2	Topic #3	Topic #4
turn	train	woods	uncle	white
wife	worker	unexpected	vacation	year
town	trapped	turns	things	troubled
true	teenagers	working	victims	trip
turned	visit	wealthy	younger	writer
perfect	tragic	university	victim	wants
survive	wrong	teacher	time	cluster_kmeans3
music	work	video	want	unknown
poor	teenager	works	team	young
suspect	truth	thriller	years	zombie
murdered	york	violent	taking	track
popular	takes	tells	travel	trouble
society	woman	travels	self	village
special	my_kmean3	thief	race	vampire
problems	undercover	weekend	person	tale

K-means clustering highlighted three distinct clusters, with one cluster showing much more variability than the other two. This model had a low silhouette score, making it less than ideal to be used for predicting and recommending movies.

While LDA is effective at providing a general idea of the top features per topic, it was unfortunately not able to provide definitive genres classifications. This is likely due to the overlap between the text descriptions and the fact that movies can belong to multiple distinct genres. Therefore, LDA may not be the most adequate model to use for a recommendation system.

Conclusion:

The models have shown that genre identifiers can indeed overlap, and that the description data may not be enough to correctly predict and suggest a movie genre matching a viewer's preferences. Creating a consistently reliable recommendation system is also made more difficult since a particular movie can fit into multiple genres, posing a challenge when aiming to associate the movie with one singular genre. Thus, it is crucial to further refine the recommendation system and ensure that it generates precise suggestions by integrating additional details like movie ratings, cast/crew information, and user activity.

Reflection & Learning Goals

The supervised machine learning methods provided a high-level overview of the strengths and limitations of each approach. While each model offers unique insight into the dataset, they collectively highlight the complexity behind recommending movie genres based solely on movie descriptions.

Results from each model highlight the performance of techniques used in the analysis, with SVM demonstrating a superior ability to classify movie genres accurately. However, the challenge of genre overlap and the multifaceted nature of movie descriptions underscore the limitations of relying exclusively on text mining for recommendation purposes. The experimentation with K-Means Clustering and LDA further emphasize the difficulty of capturing the nuances of viewer preferences through unsupervised learning models.

The skills learned and utilized from this project include:

- Collect, store, and access data.
- The application of advanced Python libraries such as pandas, NumPy, matplotlib, seaborn, sklearn, and NLTK.
- Visualization of data frames to attain valuable and intuitive insights.
- The strengths and weaknesses of supervised and unsupervised machine learning.
- Interpretation of analytical results.

Conclusion

The Applied Data Science Master’s Degree program has given me the opportunity to gain a well-round understanding of the uses and applications of data in various industries. Throughout the coursework and projects presented, I used data related to the realms of sports, finance, medicine, and entertainment. I also gained invaluable knowledge revolving around data science techniques, models, and programming language fluency that have prepared me for this competitive industry. I would like to sincerely thank the instructors, advisors, and classmates of the School of Information Studies that have accompanied me on my journey at Syracuse University.

References

"Applied Data Science Master's Degree - iSchool: Syracuse University." *Syracuse School of Information Studies*, Syracuse University, 17 June 2020, ischool.syr.edu/academics/applied-data-science-masters-degree/.

Larxel. "Heart Failure Prediction." *Kaggle*, Version 1. Retrieved 4 Feb. 2020, Accessed 6 Nov. 2022. *https://www.kaggle.com/datasets/andrewmvd/heart-failure-clinical-data*

Raju, Chidambara G. "IMDb Movie Dataset: All Movies by Genre." *Kaggle*, Version 3. Retrieved 17 February 2023, Accessed 24 Feb. 2024. *https://www.kaggle.com/datasets/rajugc/imdb-movies-dataset-based-on-genre*

"Roster Resource - Free Agent Tracker." *FanGraphs*, FanGraphs Baseball, 30 June 2022. *https://www.fangraphs.com/roster-resource/free-agent-tracker?&season=2022*

Wang, Man, and Lauren Juiliani. "IST - 707." *Answers*, Syracuse University, 16 Nov. 2021, answers.syr.edu/display/ischool/IST+707+-+Applied+Machine+Learning.

Wang, Man, and Margaret Craft. "IST 659 - Database Administration Concepts & Database Management." *Answers*, Syracuse University, 29 Apr. 2021, answers.syr.edu/pages/viewpage.action?pageId=105104343.

Wang, Man, and Margaret Craft. "IST 736 - Text Mining." *Answers*, Syracuse University, 19 Apr. 2021, answers.syr.edu/display/ischool/IST+736+-+Text+Mining.

Gianni C. Conde

(201) 820-5592 | gianniconde96@gmail.com |San Antonio, TX 78261 | linkedin.com/in/gianniconde |
github.com/gianniconde

PROFESSIONAL SUMMARY

An innovative, analytically driven Data Scientist with a strong background in applied mathematics and business analytics with a penchant for intellectual curiosity. Perceptive at providing relevant insights through data analytics, science, and visualization. Key strengths lie in manufacturing a focused understanding of business needs through strong interpersonal skills while providing adaptable solutions backed by data analysis. Self-starter with exceptional communication abilities that allow for adept collaboration with team members to arrive at common goals.

EDUCATION

Syracuse University - Syracuse, NY June 2024
Master of Science in Applied Data Science, GPA: 3.6/4.0
Shippensburg University - Shippensburg, PA May 2021
Bachelor of Science in Mathematics, Minor in Business

SKILLS

Hard Skills

- Python
- R
- SQL
- Microsoft Excel
- MongoDB
- PySpark
- CaseWare IDEA
- Google Analytics

Soft Skills

- Interpersonal Tact
- Time Management
- Adaptability
- Problem Solving
- Organization
- Emotional Intelligence
- Perseverance
- Integrity
- Dependability
- Self-motivating

PROJECTS

MLB Database Analysis, Spring 2022
Created a relational database of free agent-eligible MLB players following the 2021 season. Analyzed statistics, salaries, and other metrics to determine their influence on potential contracts and statistics for the upcoming season. Uncovered business applicable insights and trends in conjunction with a set of business rules. Language(s) used: SQL.

Heart Failure Research, Winter 2022
Examined anonymous medical data regarding patients suffering from heart failure in 2015 to discover trends related to patient survival and death. Cleaned and prepped data to uncover insights based on visualizations. Created training and testing data for supervised machine learning models, such as random forest, support vector machines, and decision trees. Utilized association rule mining to identify common instances leading to the event of death. Language(s) used: R.

Movie Recommender, Spring 2024
Examined IMDB movie text data to develop a movie recommendation model based on genre and description. Cleaned and prepped data for exploration and visualization that determined the most popular genres. Created training and testing data to develop supervised and unsupervised machine learning models. Supervised models included various naive Bayes techniques and support vector machines. Unsupervised models included k-means clustering and latent Dirichlet allocation. Language(s) used: Python.

EXPERIENCE

Primrose School at Cibolo Canyons; San Antonio, TX July 2022 - Present

Pre-Kindergarten Lead Teacher

- Taught basic skills, such as shapes, colors, letters, numbers, measurements, nutrition, hygiene, and social-emotional skills in a safe, interactive, and inclusive environment.
- Planned and implemented curriculum through conventional and technological methods alongside staff.
- Conducted regular meetings with parents to communicate child development progress, educational milestones, and individual learning strategies.