

Human Resources

Analysis Case

Team 2



- 01 - Introduction
- 02 - Data Cleaning
- 03 - SQL insights
- 04 - Machine Learning
- 05 - Visualization
- 06 - Recommendation

Our Management Team



Mahmoud Kamal



Saad Naeem



Mahmoud Gomma



Omar Ali Eldeen



Magdy Ezzat

O1 - Introduction

O2 - Data Cleaning
O4 - Machine Learning

O3 - SQL insights

O6 – Visualization

O6 – Visualization

O1 – Introduction

Objective:

This analysis was conducted on human resources data to transform it into insights and recommendations that assist decision-makers in making the right decisions.





Data sources:

HR Data Analysis CSV

Employee

Employee
Name
Gender
Age
Department
DistanceFromHome_KM
State
Ethnicity
Education
EducationField
JobRole
MaritalStatus
BusinessTravel
Salary
StockOptionLevel
OverTime
HireDate
Attrition
YearsAtCompany
YearsInCurrentRole
YearsInLastRole
YearsWithCurrManager

Performance

Performance
PerformanceID
EmployeeID
ReviewDate
EnvironmentSatisfaction
JobSatisfaction
RelationshipSatisfaction
TrainingOpportunitiesWithinYear
TrainingOpportunitiesTaken
WorkLifeBalance
SelfRating
ManagerRating

Rating

Rating
Rating_ID
RatingLevel

Satisfaction

Satisfaction
Satisfaction_ID
SatisfactionLevel

O2 - Data Cleaning

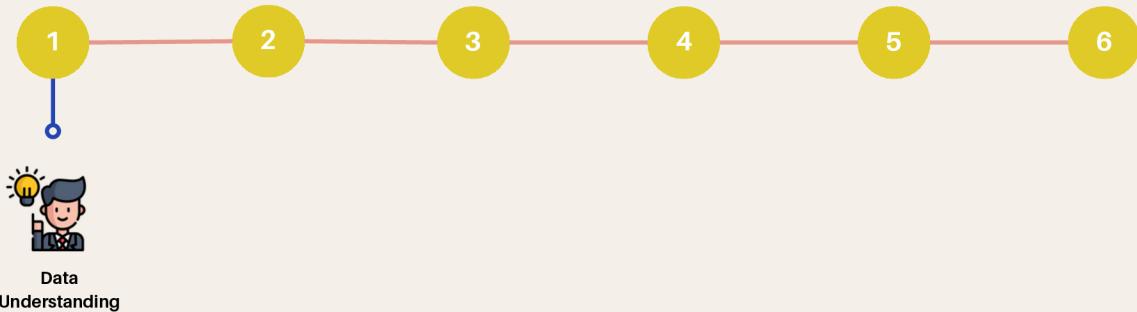
01 - Data Cleaning



Notebook Link:

https://drive.google.com/file/d/1ZTQF022ENHLacJ8CzYZjeRx2d_mo3uy1/view?usp=sharing

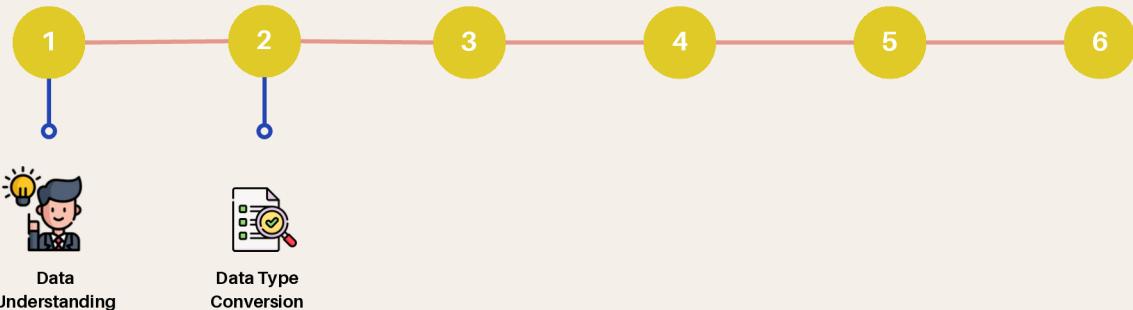
01 - Data Cleaning



Notebook Link:

https://drive.google.com/file/d/1ZTQF022ENHLacJ8CzYZjeRx2d_mo3uy1/view?usp=sharing

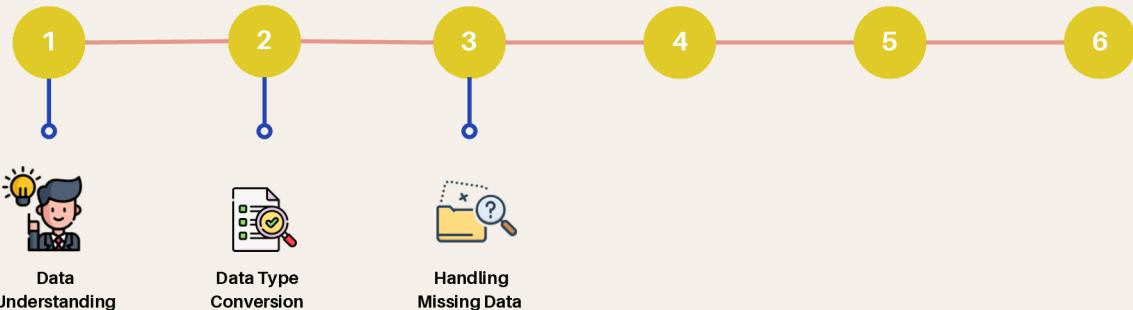
01 - Data Cleaning



Notebook Link:

https://drive.google.com/file/d/1ZTQF022ENHLacJ8CzYZjeRx2d_mo3uy1/view?usp=sharing

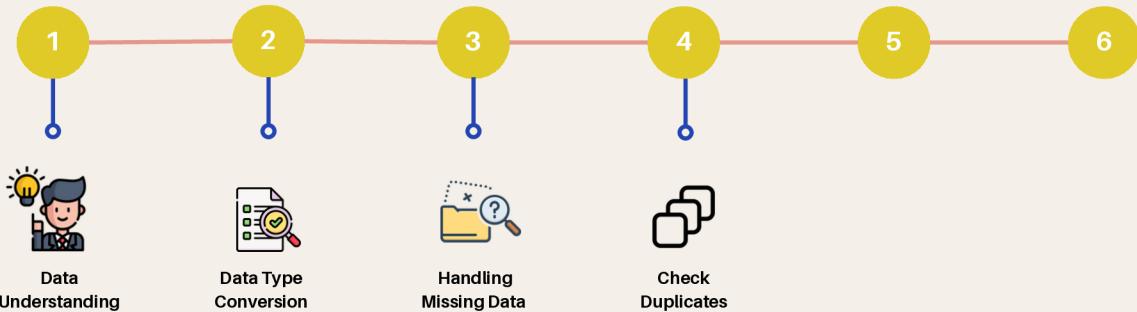
01 - Data Cleaning



Notebook Link:

https://drive.google.com/file/d/1ZTQF022ENHLacJ8CzYZjeRx2d_mo3uy1/view?usp=sharing

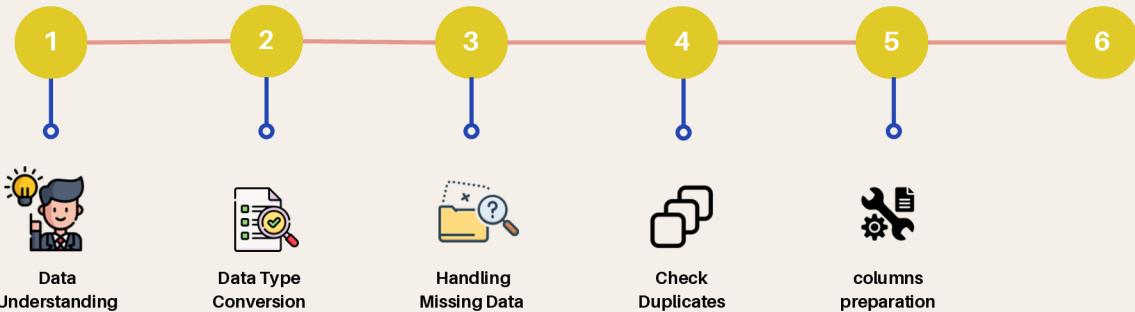
01 - Data Cleaning



Notebook Link:

https://drive.google.com/file/d/1ZTQF022ENHLacJ8CzYZjeRx2d_mo3uy1/view?usp=sharing

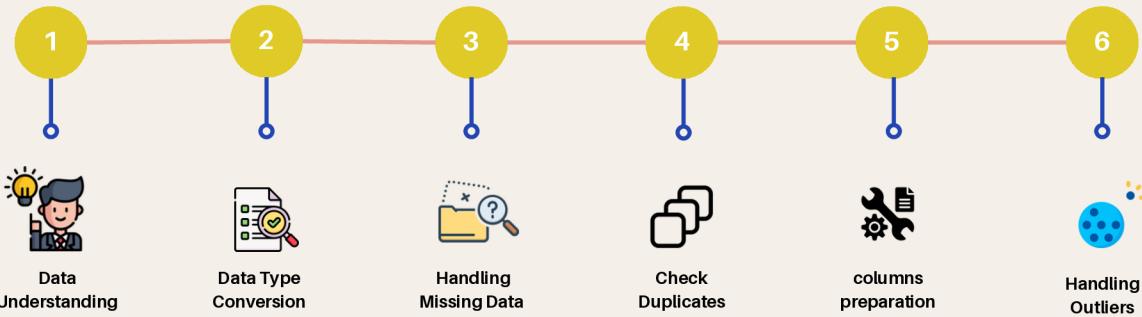
01 - Data Cleaning



Notebook Link:

https://drive.google.com/file/d/1ZTQF022ENHLacJ8CzYZjeRx2d_mo3uy1/view?usp=sharing

01 - Data Cleaning



Notebook Link:

https://drive.google.com/file/d/1ZTQF022ENHLacJ8CzYZjeRx2d_mo3uy1/view?usp=sharing

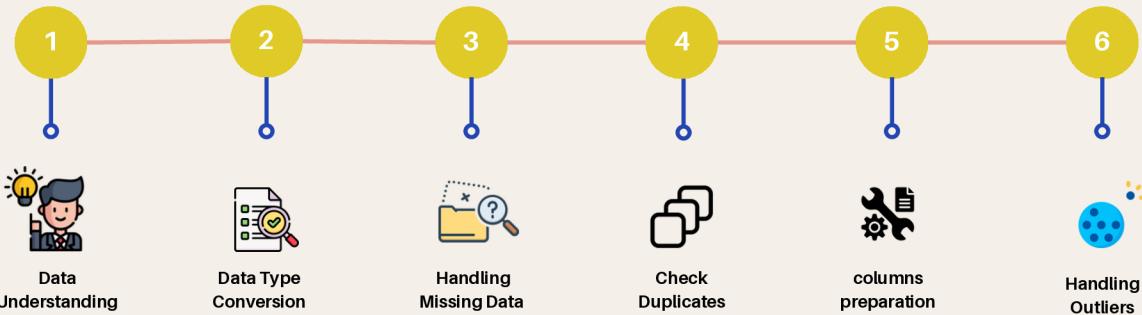
6



289768.625

Handling Outliers

01 - Data Cleaning



Notebook Link:

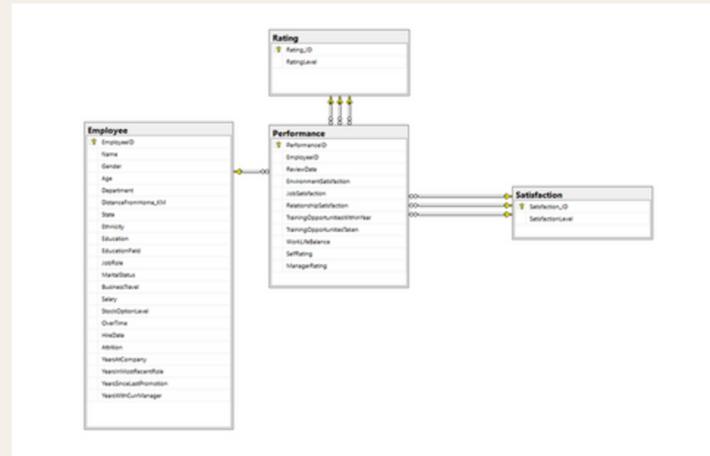
https://drive.google.com/file/d/1ZTQF022ENHLacJ8CzYZjeRx2d_mo3uy1/view?usp=sharing

O3 - SQL insights





Data Normalization



O2 - SQL insights



O2 - SQL insights



**Are the outliers
valid ?**



O2 - SQL insights

Are the outliers
valid ?



why the company have
Attrition ?



Outliers

SQL Query :

```
--1- number of outliers
SELECT
COUNT(*) as "number of outliers"
FROM Employee
WHERE Employee.Salary >= 289768.625;
```

	number of outliers
1	124

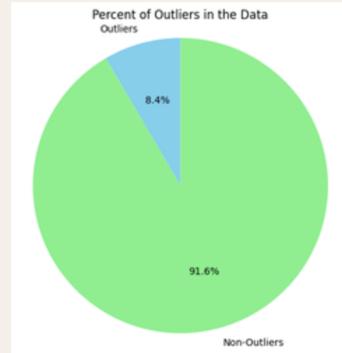
SQL Queries Link:

https://drive.google.com/file/d/1vSI5Rmq_rwEkBNLdFEg6f_nW6ul7Gqvi/view?usp=sharing

Outliers

SQL Query :

```
--2- percent of outliers in the data
SELECT
COUNT(CASE WHEN Employee.Salary >= 289768.625 THEN 1 END) * 100.0 / COUNT(*) AS "percent of outliers"
FROM Employee;
```

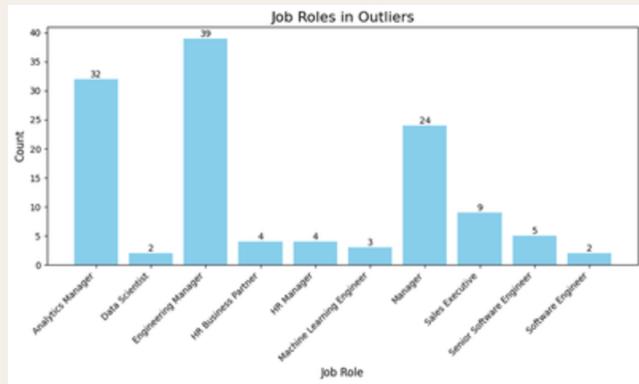


Python Visualization

Outliers

SQL Query :

```
--3- what is jobRole for outliers
SELECT
JobRole, COUNT(*) AS "count"
FROM Employee
WHERE Salary > 289768.625
GROUP BY JobRole;
```



Python Visualization

Outliers

SQL Query :

```
--4- manager salary in outliers
SELECT
JobRole, COUNT(*) as "Count of JobRoles"
FROM Employee
WHERE Salary > 289768.625 AND JobRole LIKE '%Manager%'
GROUP BY JobRole;
```

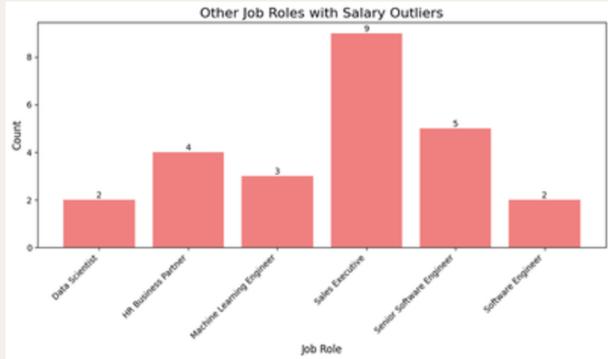


Python Visualization

Outliers

SQL Query :

```
--5- aother salary in outliers
SELECT
JobRole, COUNT(*) as "Count of JobRoles"
FROM Employee
WHERE Salary > 289768.625 AND JobRole not LIKE '%Manager%'
GROUP BY JobRole;
```



Python Visualization

Outliers

SQL Query :

```
--> percent of manager and others in outliers
SELECT
    COUNT(CASE WHEN Employee.JobRole LIKE '%Manager%' AND Employee.Salary > 289768.625 THEN 1 END) * 100.0 /
    COUNT(CASE WHEN Employee.Salary > 289768.625 THEN 1 END) AS "Percent of manager in outliers",
    COUNT(CASE WHEN Employee.JobRole NOT LIKE '%Manager%' AND Employee.Salary > 289768.625 THEN 1 END) * 100.0 /
    COUNT(CASE WHEN Employee.Salary > 289768.625 THEN 1 END) AS "Percent of not manager in outliers"
FROM
    Employee;
```



Python Visualization

Outliers

SQL Query :

```
--7- manager by Age, Education and YearsAtCompany  
select  
Distinct JobRole,Age,salary,Employee.Education,Employee.YearsAtCompany  
from Employee join Performance  
on Employee.EmployeeID = Performance.EmployeeID  
where Salary > 289768.625 and Employee.JobRole LIKE '%Manager%'  
group by JobRole,Age,salary,Employee.Education,Employee.YearsAtCompany,YearsInMostRecentRole, Employee.YearsSinceLastPromotion  
order by Age DESC
```

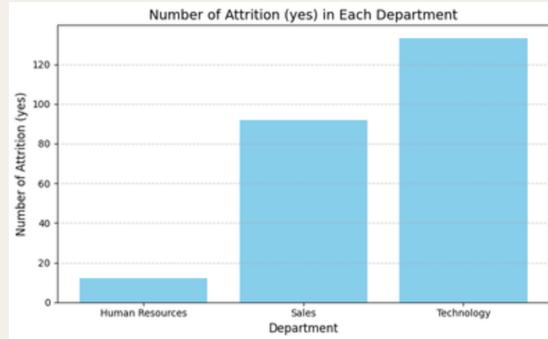


Python Visualization

ATTRITION

SQL Query :

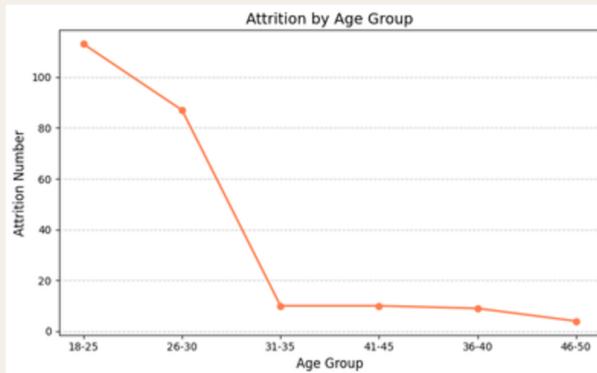
```
--1- relation between Attrition and Department
SELECT
    Employee.Department,
    COUNT(CASE WHEN EmployeeAttrition = 'yes' THEN 1 END) AS "Attrition Number",
    COUNT(CASE WHEN EmployeeAttrition = 'no' THEN 1 END) AS "Non-Attrition Number",
    COUNT(*) AS "Total Employees",
    COUNT(CASE WHEN EmployeeAttrition = 'yes' THEN 1 END) * 100.0 / COUNT(*) AS "Attrition Percentage",
    COUNT(CASE WHEN EmployeeAttrition = 'no' THEN 1 END) * 100.0 / COUNT(*) AS "Non-Attrition Percentage"
FROM
    Employee
GROUP BY
    Employee.Department
ORDER BY
    Employee.Department;
```



Python Visualization

ATTRITION

```
--2- What is the relationship between 'Age' and 'Attrition'?
SELECT
    CASE
        WHEN Employee.Age BETWEEN 18 AND 25 THEN '18-25'
        WHEN Employee.Age BETWEEN 26 AND 30 THEN '26-30'
        WHEN Employee.Age BETWEEN 31 AND 35 THEN '31-35'
        WHEN Employee.Age BETWEEN 36 AND 40 THEN '36-40'
        WHEN Employee.Age BETWEEN 41 AND 45 THEN '41-45'
        WHEN Employee.Age BETWEEN 46 AND 50 THEN '46-50'
    END AS Age_Group,
    COUNT(Employee.Attrition) AS "Attrition Number"
FROM Employee
WHERE Employee.Attrition = 'yes'
GROUP BY
    CASE
        WHEN Employee.Age BETWEEN 18 AND 25 THEN '18-25'
        WHEN Employee.Age BETWEEN 26 AND 30 THEN '26-30'
        WHEN Employee.Age BETWEEN 31 AND 35 THEN '31-35'
        WHEN Employee.Age BETWEEN 36 AND 40 THEN '36-40'
        WHEN Employee.Age BETWEEN 41 AND 45 THEN '41-45'
        WHEN Employee.Age BETWEEN 46 AND 50 THEN '46-50'
    END
ORDER BY "Attrition Number" DESC;
```



Python Visualization

ATTRITION

```
--3. What is the relationship between 'Salary' and 'Attrition'?  
SELECT  
    CASE  
        WHEN Employee.Salary BETWEEN 20000 AND 50000 THEN '20,000-50,000'  
        WHEN Employee.Salary BETWEEN 50001 AND 100000 THEN '50,001-100,000'  
        WHEN Employee.Salary BETWEEN 100001 AND 150000 THEN '100,001-150,000'  
        WHEN Employee.Salary BETWEEN 150001 AND 200000 THEN '150,001-200,000'  
        WHEN Employee.Salary BETWEEN 200001 AND 250000 THEN '200,001-250,000'  
        WHEN Employee.Salary BETWEEN 250001 AND 300000 THEN '250,001-300,000'  
        ELSE 'Above 300,000'  
    END AS Salary_Range,  
    COUNT(Employee.Attrition) AS "Attrition Number"  
FROM  
    Employee  
WHERE  
    Employee.Attrition = 'yes'  
GROUP BY  
    CASE  
        WHEN Employee.Salary BETWEEN 20000 AND 50000 THEN '20,000-50,000'  
        WHEN Employee.Salary BETWEEN 50001 AND 100000 THEN '50,001-100,000'  
        WHEN Employee.Salary BETWEEN 100001 AND 150000 THEN '100,001-150,000'  
        WHEN Employee.Salary BETWEEN 150001 AND 200000 THEN '150,001-200,000'  
        WHEN Employee.Salary BETWEEN 200001 AND 250000 THEN '200,001-250,000'  
        WHEN Employee.Salary BETWEEN 250001 AND 300000 THEN '250,001-300,000'  
        ELSE 'Above 300,000'  
    END  
ORDER BY  
    "Attrition Number" DESC;
```

output:

	Salary Range	Attrition Number
1	20,000-50,000	117
2	50,001-100,000	67
3	100,001-150,000	21
4	200,001-250,000	10
5	150,001-200,000	9
6	Above 300,000	8
7	250,001-300,000	5

ATTRITION

SQL Query :

```
--4- manager rating and Attrition (percentage)
SELECT
    P.ManagerRating,
    COUNT(CASE WHEN E.Attrition = 'yes' THEN 1 END) * 100.0 / COUNT(*) AS "Attrition Percentage",
    COUNT(CASE WHEN E.Attrition = 'no' THEN 1 END) * 100.0 / COUNT(*) AS "Non-Attrition Percentage"
FROM
    Employee E
JOIN
    Performance P ON E.EmployeeID = P.EmployeeID
GROUP BY
    P.ManagerRating
ORDER BY
    P.ManagerRating;
```

output:

	ManagerRating	Attrition Percentage	Non-Attrition Percentage
1	2	34.731543624161	65.268456375838
2	3	33.783175888439	66.216824111560
3	4	32.927927927927	67.072072072072
4	5	33.985102420856	66.014897579143

ATTRITION

SQL Query :

```
--S- 'satisfaction-Level' and 'Attrition' (percentage)
SELECT
    S.SatisfactionLevel,
    P.JobSatisfaction,
    COUNT(CASE WHEN E.Attrition = 'yes' THEN 1 END) AS "Attrition Number",
    COUNT(CASE WHEN E.Attrition = 'no' THEN 1 END) AS "Non-Attrition Number",
    COUNT(*) AS "Total Employees",
    COUNT(CASE WHEN E.Attrition = 'yes' THEN 1 END) * 100.0 / COUNT(*) AS "Attrition Percentage", -- لـ ١٠٠٪ احتفظ
    COUNT(CASE WHEN E.Attrition = 'no' THEN 1 END) * 100.0 / COUNT(*) AS "Non-Attrition Percentage" -- لـ ١٠٠٪ احتفظ
FROM
    Employee E
JOIN
    Performance P ON E.EmployeeID = P.EmployeeID
JOIN
    Satisfaction S ON S.Satisfaction_ID = P.JobSatisfaction
GROUP BY
    S.SatisfactionLevel, P.JobSatisfaction
ORDER BY
    P.JobSatisfaction, S.SatisfactionLevel;
```

output:

	SatisfactionLevel	JobSatisfaction	Attrition Number	Non-Attrition Number	Total Employees	Attrition Percentage	Non-Attrition Percentage
1	Very Dissatisfied	1	36	94	130	27.692307692307	72.307692307692
2	Dissatisfied	2	549	1125	1674	32.795658924731	67.204301075268
3	Neutral	3	568	1083	1651	34.403391883706	65.596608116293
4	Satisfied	4	573	1112	1685	34.005934718100	65.994065281899
5	Very Satisfied	5	535	1034	1569	34.098151688973	65.901848311026

ATTRITION

SQL Query :

```
--6- Environment-Satisfaction and 'Attrition' ( percentage )
SELECT
    P.EnvironmentSatisfaction,
    COUNT(CASE WHEN E.Attrition = 'yes' THEN 1 END) AS "Attrition Number",
    COUNT(CASE WHEN E.Attrition = 'no' THEN 1 END) AS "Non-Attrition Number",
    COUNT(*) AS "Total Employees",
    COUNT(CASE WHEN E.Attrition = 'yes' THEN 1 END) * 100.0 / COUNT(*) AS "Attrition Percentage",
    COUNT(CASE WHEN E.Attrition = 'no' THEN 1 END) * 100.0 / COUNT(*) AS "Non-Attrition Percentage"
FROM
    Employee E
JOIN
    Performance P ON E.EmployeeID = P.EmployeeID
GROUP BY
    P.EnvironmentSatisfaction
ORDER BY
    P.EnvironmentSatisfaction ASC;
```

output:

	EnvironmentSatisfaction	Attrition Number	Non-Attrition Number	Total Employees	Attrition Percentage	Non-Attrition Percentage
1	1	35	101	136	25.735294117647	74.264705882352
2	2	44	97	141	31.205673758865	68.794326241134
3	3	776	1435	2211	35.097241067390	64.902758932609
4	4	706	1469	2175	32.459770114942	67.540229885057
5	5	700	1346	2046	34.213098729227	65.786901270772

ATTRITION

SQL Query :

```
--7-- Years At company and Attrition ( percentage )  
SELECT  
    E.YearsAtCompany,  
    COUNT(CASE WHEN E.Attrition = 'yes' THEN 1 END) AS "Attrition Number",  
    COUNT(CASE WHEN E.Attrition = 'no' THEN 1 END) AS "Non-Attrition Number",  
    COUNT(*) AS "Total Employees",  
    COUNT(CASE WHEN E.Attrition = 'yes' THEN 1 END) * 100.0 / COUNT(*) AS "Attrition Percentage",  
    COUNT(CASE WHEN E.Attrition = 'no' THEN 1 END) * 100.0 / COUNT(*) AS "Non-Attrition Percentage"  
FROM  
    Employee E  
GROUP BY  
    E.YearsAtCompany  
ORDER BY  
    E.yearsAtCompany ASC;
```

output:

	YearsAtCompany	Attrition Number	Non-Attrition Number	Total Employees	Attrition Percentage	Non-Attrition Percentage
1	0	60	130	190	31.578947368421	68.421052631578
2	1	61	116	177	34.463276836158	65.536723163841
3	2	25	99	124	20.161290322580	79.838709677419
4	3	24	124	148	16.216216216216	83.783783783783
5	4	15	114	129	11.627906976744	88.372093023255
6	5	20	95	115	17.391304347826	82.608695652173
7	6	11	90	101	10.891089108910	89.108910891089
8	7	9	112	121	7.438016528925	92.561983471074
9	8	6	113	119	5.042016806722	94.957983193277
10	9	5	113	118	4.237288135593	95.762711864406
11	10	1	127	128	0.781250000000	99.218750000000

ATTRITION

SQL Query :

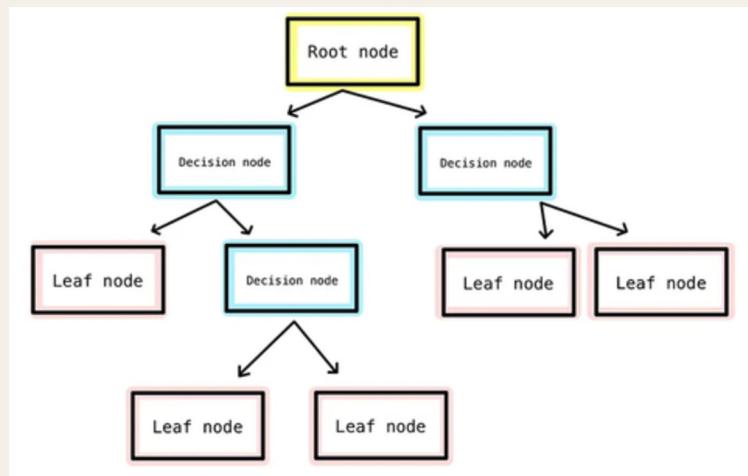
```
--8- job role , 'attrition' ( percentage )
SELECT
    Employee.JobRole,
    COUNT(CASE WHEN Employee.Attrition = 'yes' THEN 1 END) AS "Attrition Number",
    COUNT(CASE WHEN Employee.Attrition = 'No' THEN 1 END) AS " Non Attrition Number",
    COUNT(*) AS "Total Employees",
    COUNT(CASE WHEN Employee.Attrition = 'yes' THEN 1 END) * 100.0 / COUNT(*) AS "Attrition Percentage"
FROM
    Employee
GROUP BY
    Employee.JobRole
ORDER BY
    "Attrition Percentage" DESC;
```

output:

JobRole	Attrition Number	Non Attrition Number	Total Employees	Attrition Percentage
1 Sales Representative	33	50	83	39.759036144578
2 Recruiter	9	15	24	37.500000000000
3 Data Scientist	62	199	261	23.754789272030
4 Sales Executive	57	270	327	17.431192660550
5 Software Engineer	47	247	294	15.986394557823
6 HR Executive	3	25	28	10.714285714285
7 Machine Learning Engineer	10	136	146	6.849315068493
8 Senior Software Engineer	9	123	132	6.818181818181
9 Analytics Manager	3	49	52	5.769230769230
10 Manager	2	35	37	5.405405405405
11 Engineering Manager	2	73	75	2.666666666666
12 HR Business Partner	0	7	7	0.000000000000
13 HR Manager	0	4	4	0.000000000000

O4 - Machine Learning

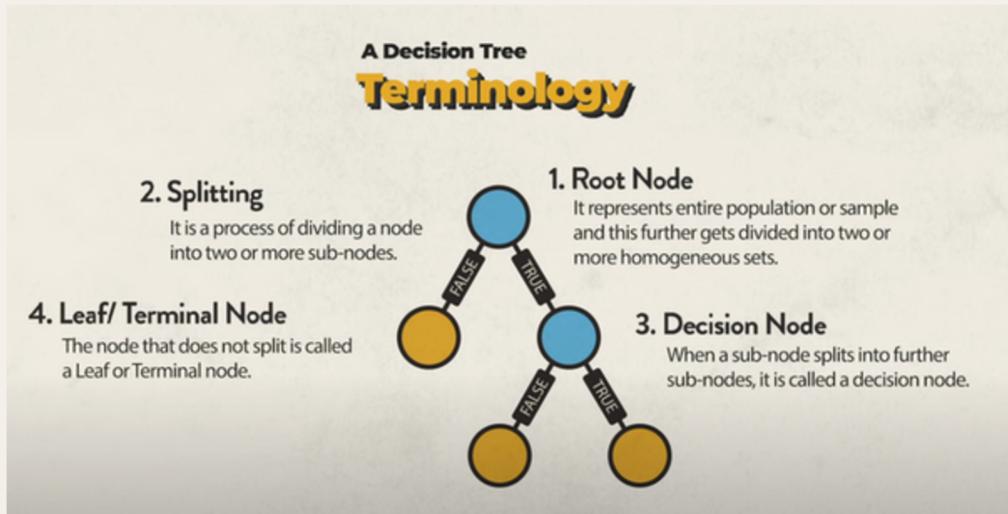
Decision Tree Classifier



Notebook Link:

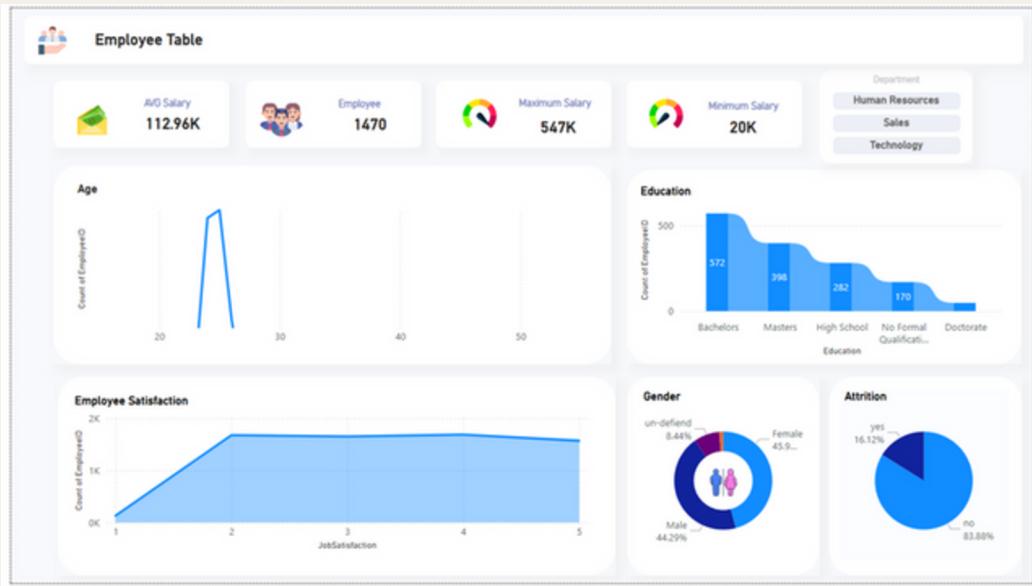
https://colab.research.google.com/drive/1hwQM6uPTjuASsXRU2saxLOZ_ydVU-iFE?usp=sharing

Decision Tree Classifier



06 - Visualization

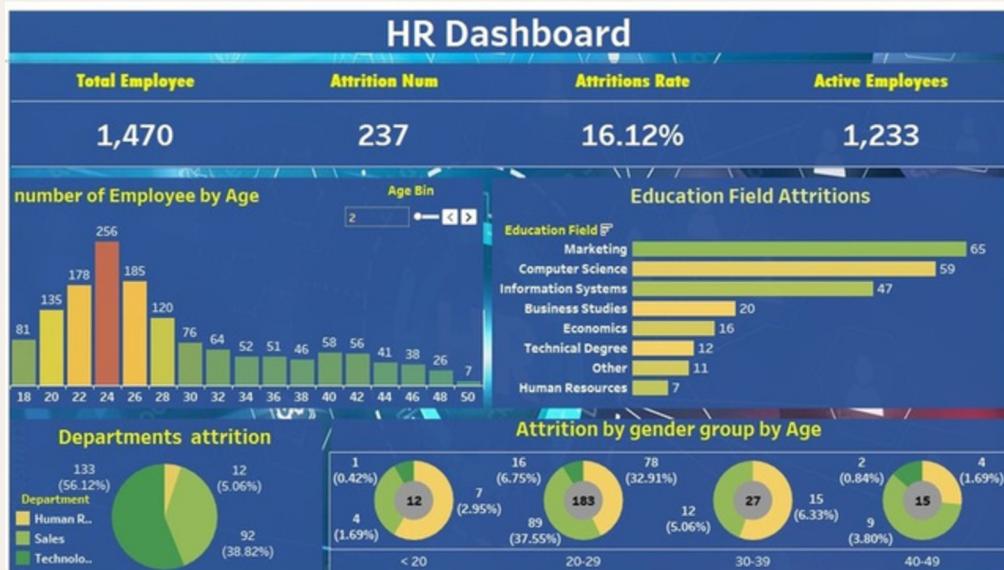
Employees Dashboard



Dashboard Link:

<https://drive.google.com/file/d/1CSGtQESFzN5pnly5FeHmletqIDPTNANA/view?usp=sharing>

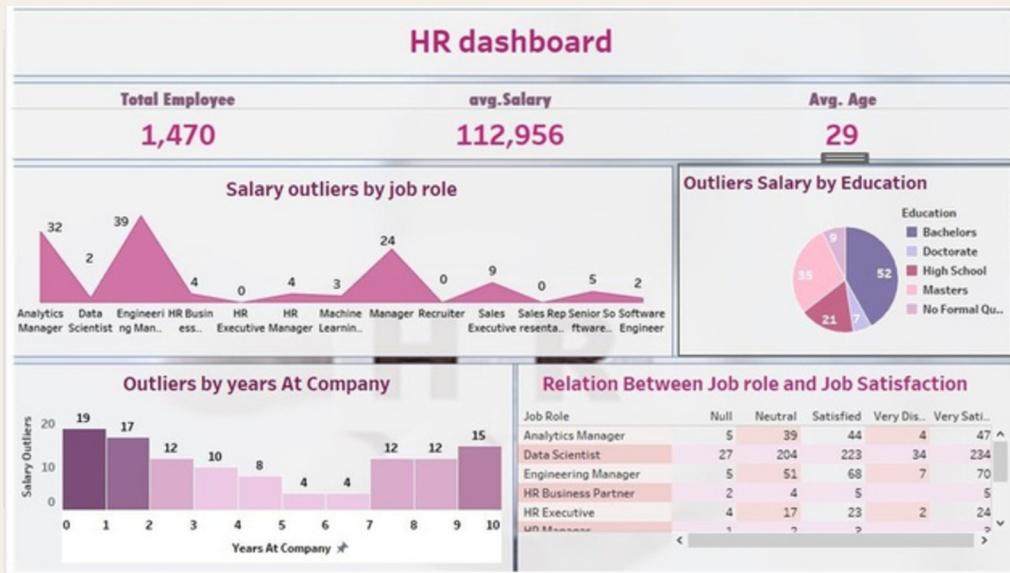
Attrition Dashboard



Dashboard Link:

<https://drive.google.com/file/d/12FXM80rrUoI3q6YeZY6xNc70ibdSUPnN/view?usp=sharing>

Outliers Dashboard



Dashboard Link:

<https://drive.google.com/file/d/12FXM80rrUoI3q6YeZY6xNc70ibdSUPnN/view?usp=sharing>

06 - Recommendation

Actions to be Taken:

- Through analysis, we were able to identify the age group most likely to experience attrition in the workplace. This group consists of young people aged 20-29, making it crucial to develop more effective attraction strategies to meet their needs and retain them.
- **Employee Engagement Initiatives:** Implement programs aimed at increasing employee engagement, such as regular feedback sessions, team-building activities, and recognition programs to boost morale and reduce attrition.
- **Department-Specific Strategies:** Conduct detailed assessments of departments with higher attrition rates to identify specific issues and tailor retention strategies accordingly, such as additional training or resources.

- **Career Development Opportunities:** Create clear pathways for career advancement and professional development to encourage employees to stay and grow within the organization.
- In addition to the job role, there are several factors to consider when determining the appropriate salary for each individual in an organization.
The two main factors are:
 - **Experience:** referred to here as the number of years in the company, should have a balance between the number of years an employee has spent in the company and their salary.
 - **The other factor is Education:** Obtaining a higher educational degree should result in a salary that reflects and values the individual's level of education.

Thanks