IBM Performance Analysis

Presented By Group WDyZnLiP4

郑慧琳 5141619006

邓迪 5141209241

夏欣羽 515010910004

张婧文 516120910184

李玥沁 516120910175

Content

Part 1

Data Description

Import the dataset

```
data = pd.read_csv("WA_Fn-UseC_-HR-Employee-Attrition.csv")
data.head()
```

Describe the statistics

```
data.describe()
```

	Age	DailyRate	DistanceFromHome	Education	EmployeeCount	Employe
count	1470.000000	1470.000000	1470.000000	1470.000000	1470.0	1470.000
mean	36.923810	802.485714	9.192517	2.912925	1.0	1024.865
std	9.135373	403.509100	8.106864	1.024165	0.0	602.0243
min	18.000000	102.000000	1.000000	1.000000	1.0	1.000000
25%	30.000000	465.000000	2.000000	2.000000	1.0	491.2500
50%	36.000000	802.000000	7.000000	3.000000	1.0	1020.500
75%	43.000000	1157.000000	14.000000	4.000000	1.0	1555.750
max	60.000000	1499.000000	29.000000	5.000000	1.0	2068.000

8 rows × 26 columns

Part 2

Data Processing

Transfer the form into int

- Code

```
data = pd.read_csv("WA_Fn-UseC_-HR-Employee-Attrition.csv")
mapping_attrition = {'Yes':1,'No':2}
data.replace({'Attrition':mapping_attrition},inplace=True)

mapping_BusinessTravel = {'Non-Travel':1, 'Travel_Rarely':2, 'Travel_Frequently':3}
data.replace({'BusinessTravel':mapping_BusinessTravel},inplace=True)

mapping_Department = {'Sales':1,'Research & Development':2,'Human Resources':3}
data.replace({'Department':mapping_Department},inplace=True)

mapping_EducationField  = {'Medical':1,'Life Sciences':2,'Human Resources':3,'Technical Degree':4,'Marketing':5,'Other':6}
data.replace({'EducationField':mapping_EducationField},inplace=True)

mapping_Gender  = {'Female':1,'Male':2}
data.replace({'Gender':mapping_Gender},inplace=True)

mapping_JobRole = {'Sales Executive':1,'Research Scientist':2,'Laboratory Technician':3,'Manufacturing Director':4,'Manager':5,'Health
data.replace({'JobRole':mapping_JobRole},inplace=True)

mapping_Over18 = {'Y':1}
data.replace({'Over18':mapping_Over18},inplace=True)

mapping_OverTime = {'Yes':1,'No':2}
data.replace({'OverTime':mapping_OverTime},inplace=True)

mapping_MaritalStatus = {'Single':1,'Married':2,'Divorced':3}
data.replace({'MaritalStatus':mapping_MaritalStatus},inplace=True)
cols=['Attrition','BusinessTravel']

data.info()
```

- Outcome

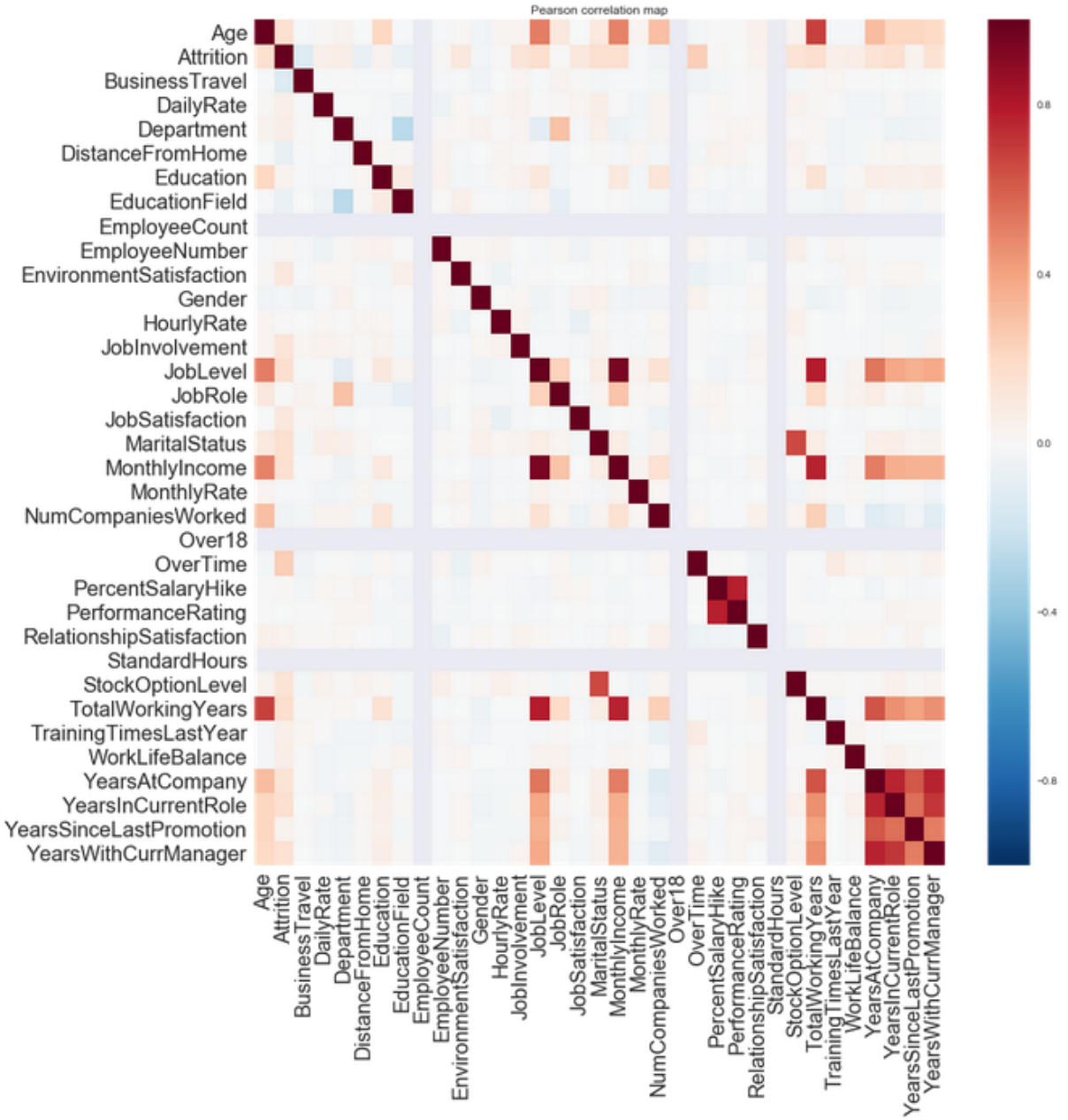
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1470 entries, 0 to 1469
Data columns (total 35 columns):
Age                       1470 non-null int64
Attrition                 1470 non-null int64
BusinessTravel            1470 non-null int64
DailyRate                 1470 non-null int64
Department                1470 non-null int64
DistanceFromHome          1470 non-null int64
Education                 1470 non-null int64
EducationField            1470 non-null int64
EmployeeCount             1470 non-null int64
EmployeeNumber            1470 non-null int64
EnvironmentSatisfaction   1470 non-null int64
Gender                    1470 non-null int64
HourlyRate                1470 non-null int64
JobInvolvement            1470 non-null int64
JobLevel                  1470 non-null int64
JobRole                   1470 non-null int64
JobSatisfaction           1470 non-null int64
```

Part 3

Visualization

Pearson Map

```
fig = plt.figure(figsize=(15, 15))
ax = fig.add_subplot(1,1,1)
corr_data = data.select_dtypes(["number"]).corr()
sns.heatmap(corr_data, ax=ax)
ax.tick_params(axis='both', which='major', labelsize=20)
ax.set_title("Pearson correlation map")
plt.tight_layout()
plt.show()
```



4 most inner-related

- Job level and monthly income;
- Age and total working years;
- Job level and total working years;
- Monthly income and total working years;

Plot histogram for our target value satisfaction

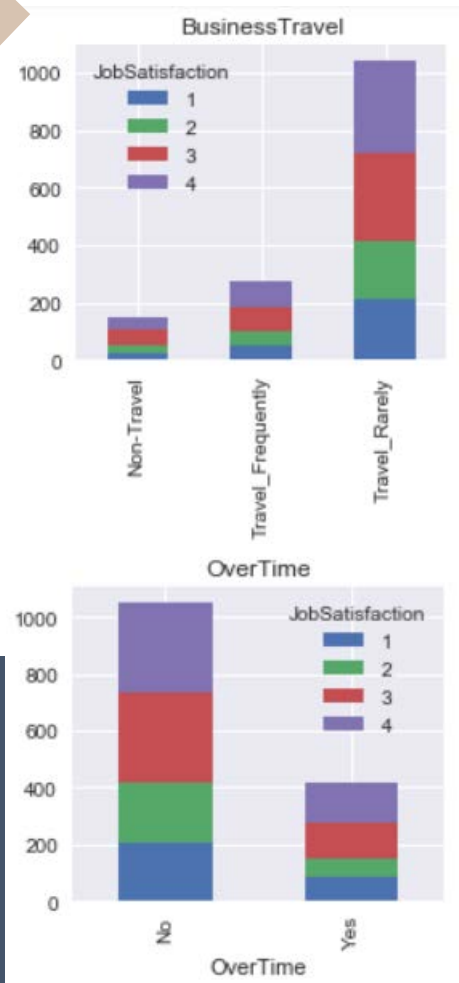
To observe the distribution of the satisfaction.

```
fig = plt.figure(figsize=(10, 52))
cols = 3
target_column = 'JobSatisfaction'
rows = np.ceil(float(data.shape[1] / cols))
a=[target_column , 'PerformanceRating' , 'EnvironmentSatisfaction' , 'RelationshipSatisfaction']
for i, column in enumerate(data.columns):
    if column in a:
        continue
    ax = fig.add_subplot(rows, cols, i+1)
    ax.set_title(column)
    if data.dtypes[column] == np.object:
        cts = data[[target_column, column]]
        cts = cts.groupby([target_column, column]).size()
        cts.unstack().T.plot(kind="bar", ax=ax, stacked=True, alpha=1)
```

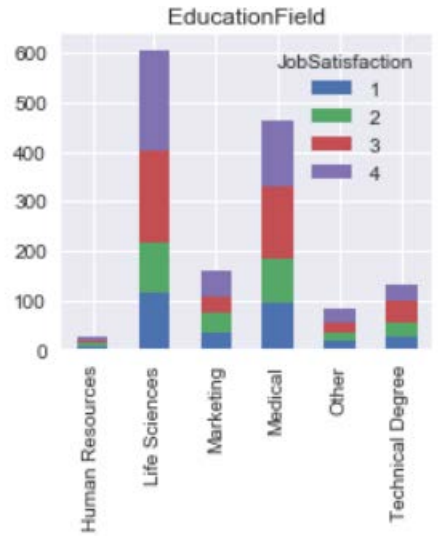
```
else:
        cts = data[[target_column, column]]
        #(xmin, xmax) = (min(cts[column].tolist()),
        max(cts[column].tolist()))

cts.groupby(target_column)[column].plot(
            bins=16,
            kind="hist",
            stacked=True,
            alpha=1,
            legend=True,
            ax=ax,
            #range=[xmin, max]
        )
plt.tight_layout()
```

Related factors



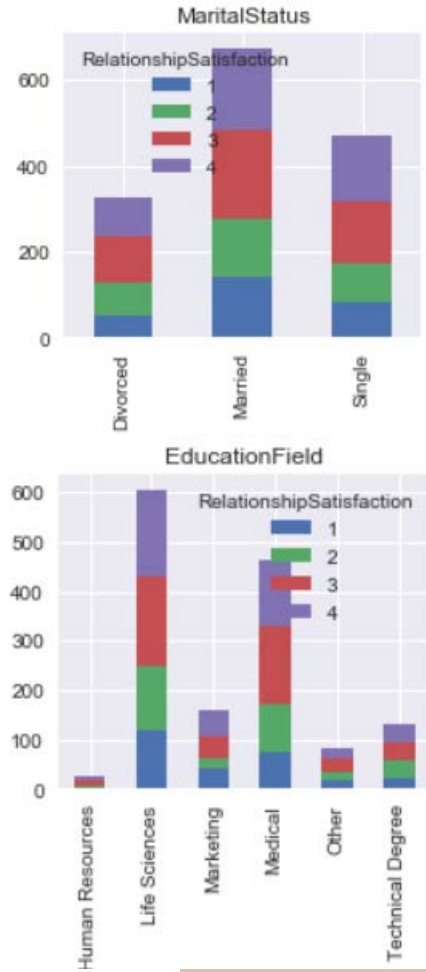
Unrelated factors



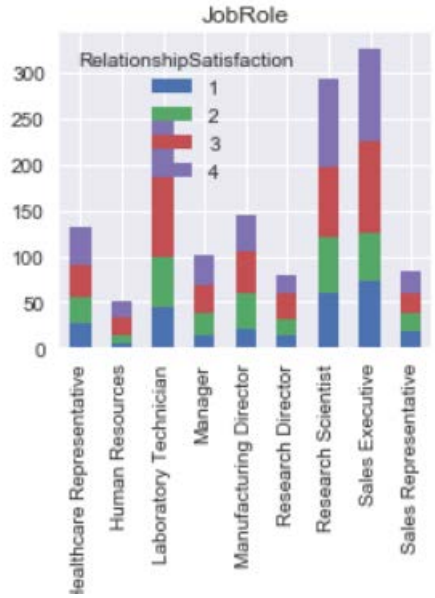
job satisfaction

Part 3| Visualization

Related factors

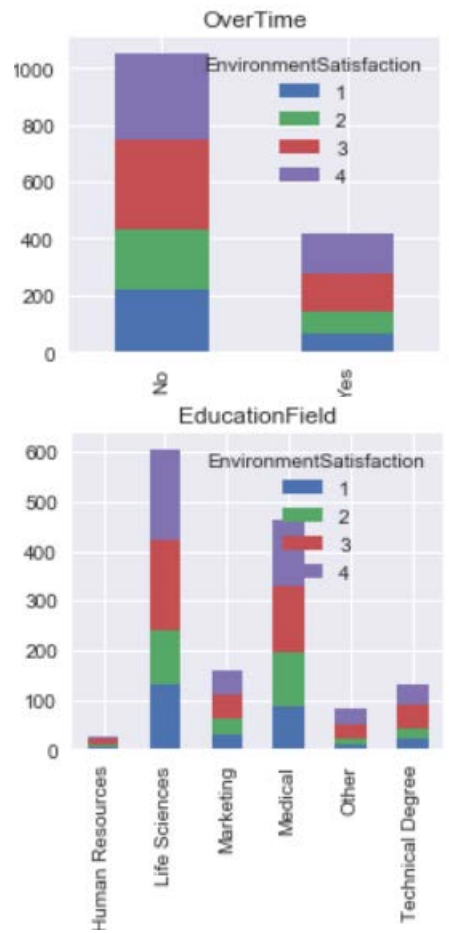


Unrelated factors

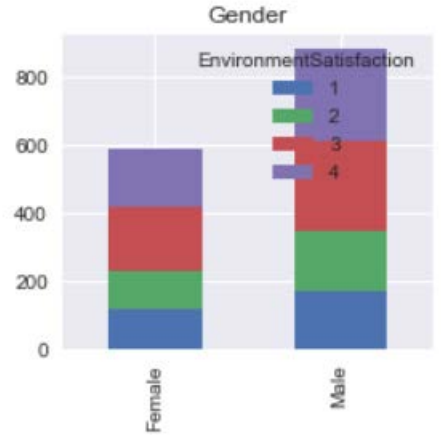


Relationship satisfaction

Related factors



Unrelated factors



Environment satisfaction

Satisfaction corelation

to find out the relationship between satisfaction and other factors

- **delete the column that has only one values for all rows**

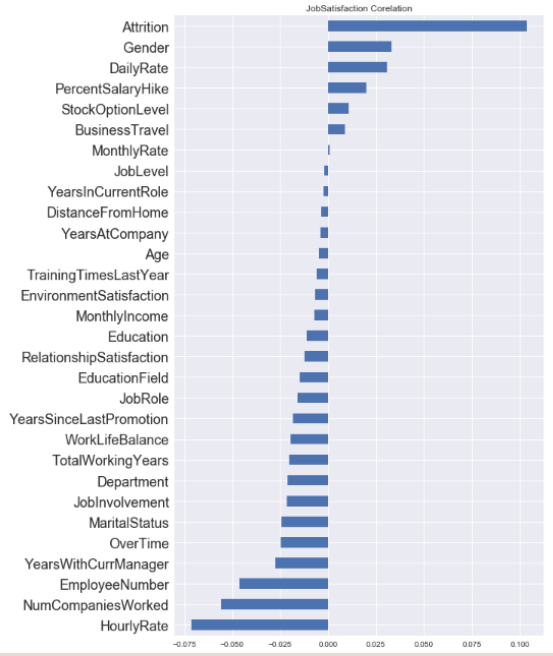
```
no_inf = uniq.index[uniq==1]
print(no_inf)
data.drop(labels=no_inf, axis=1, inplace=True)
```

```
Index(['EmployeeCount', 'Over18', 'StandardHours'],
dtype='object')
```

- **plot histogram**

```
data.drop("PerformanceRating", axis=1, inplace=True)
fig = plt.figure(figsize=(10, 10))
ax = fig.add_subplot(1,1,1)
ax =
data.corr().ix["JobSatisfaction"].drop("JobSatisfaction").
sort_values().plot(kind="barh", figsize=(10, 12), ax=ax)
ax.tick_params(axis='y', which='major', labelsize=18)
ax.set_title("JobSatisfaction Corelation")
plt.tight_layout()
#plt.savefig("JobSatisfactionCorelation.png")
```

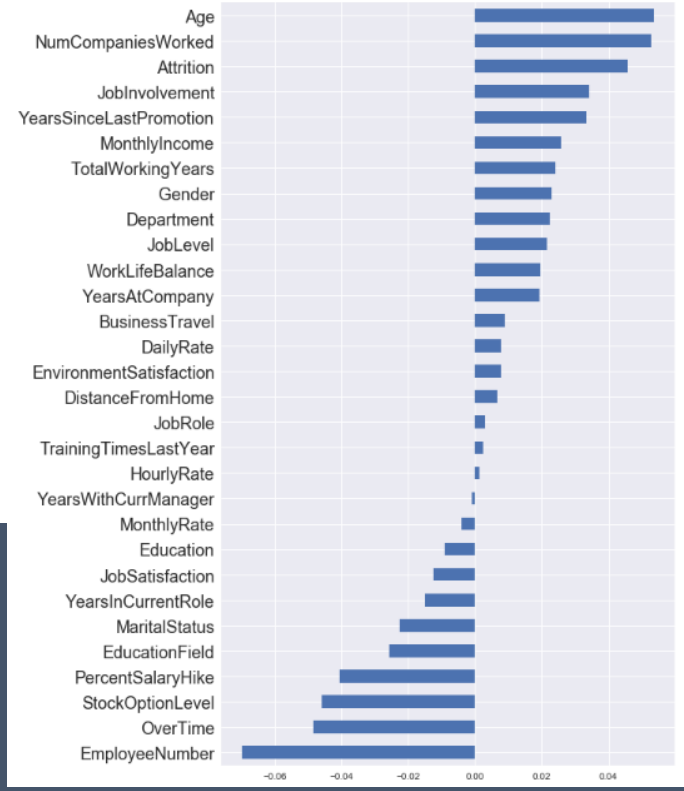
- **job satisfaction corelation**



positve:
gender, daily rate, percenta salary hike
negative:
hourly rate, num companies worked, employee number

Satisfaction corelation

- **environment satisfaction corelation**



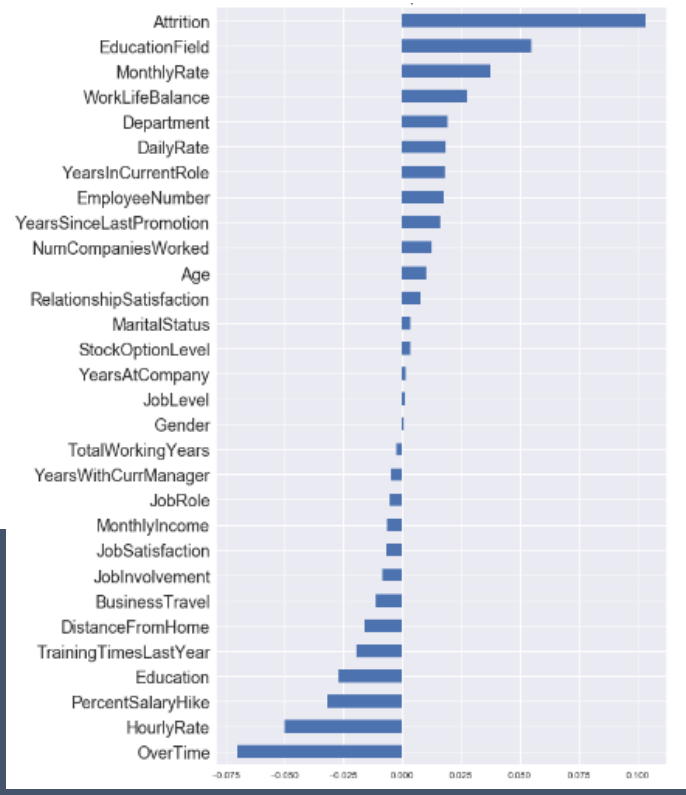
positve:

age, num corrpanies worked, job involvement

negative:

employee number, over time, stock option level

- **relationship satisfaction corelation**



positve:

education field, monthly rate, work life balance.

negative:

over time, hourly rate, percent salary hike.

IV Moderator
(satisfaction)

positive	negative
age	emplyee number
num companies worked	over time 2
job involvement	stock option level
education field	percent salary hike
monthly rate	hourly rate 2
work life balance	percent salary hike
gender	num companies worked
daily rate	
percenta salary hike	

Part 4

Machine learning

4.1

Preparation of the data

Factors:
attrition
department
educational field
gender
job role
age
time
marital status

4.2:
value 1:
key None-Business Travel
value 2:
key Travel-Rarely
value 3:
key Travel-Frequently

Machine learning method

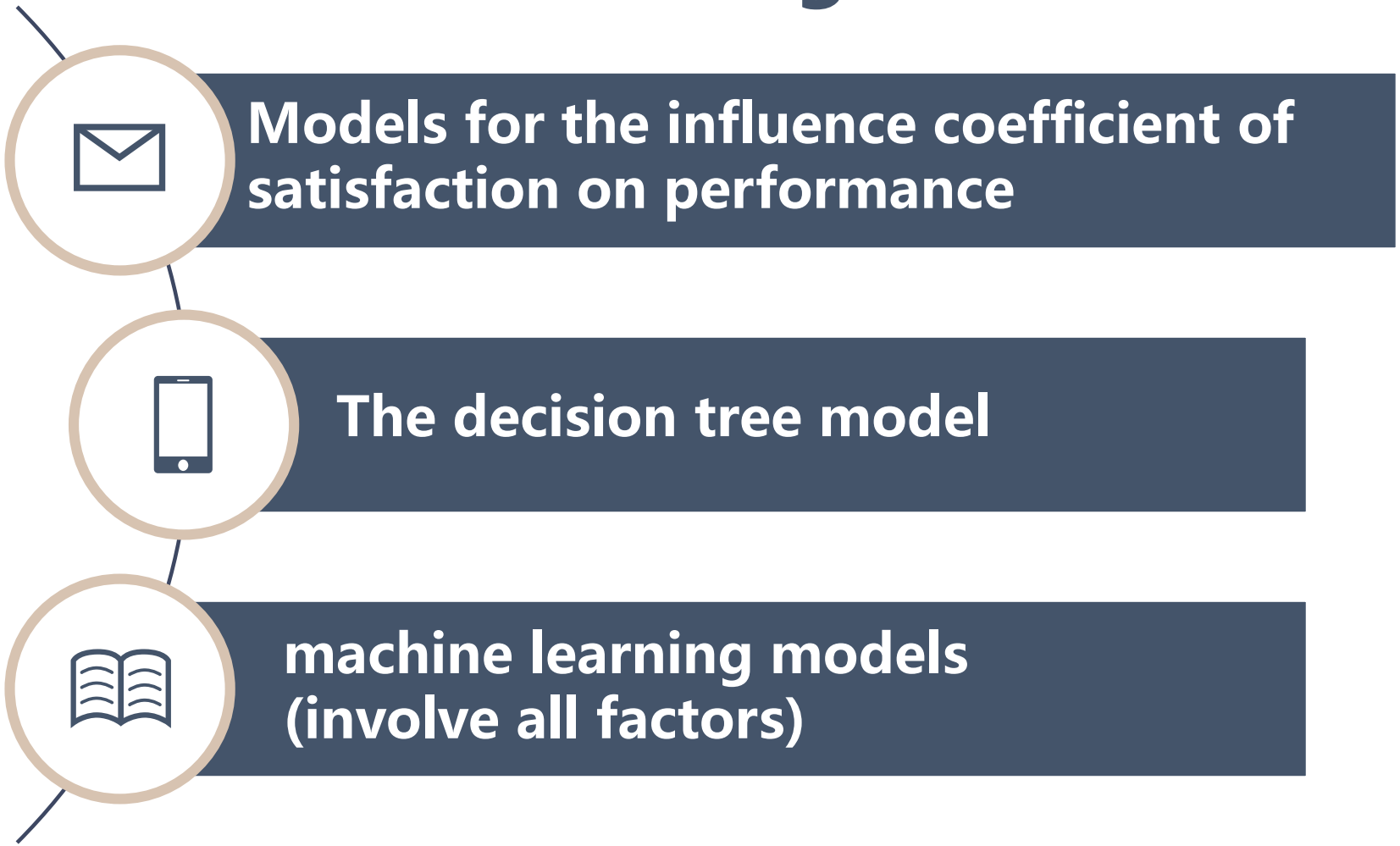
4.1

Preparation of the data

4.2

Machine learning method

Machine learning method



Machine learning method

Models for the influence coefficient of satisfaction on performance

Machine learning algorithms :

Accuracy :

- ✓ K Nearest Neighbors Regressor(KNN)
  - ➢ mean square errors
- ✓ Linear Regression
  - ➢ root mean square errors
- ✓ Decision Tree Regressor
- ✓ Random Forest Regressor

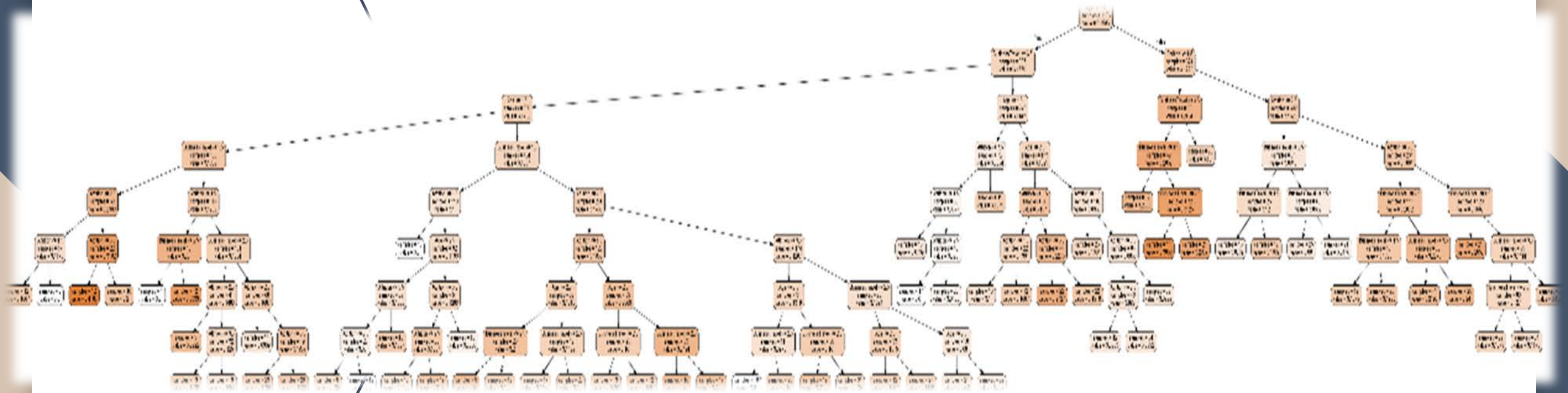
Satisfaction dimensions :

Job satisfaction

Environment satisfaction

Relationship satisfaction

Machine learning method



The decision tree

Mean Square Error (TREE): 0.15151289157

Root Mean Square Error (TREE): 0.389246569118

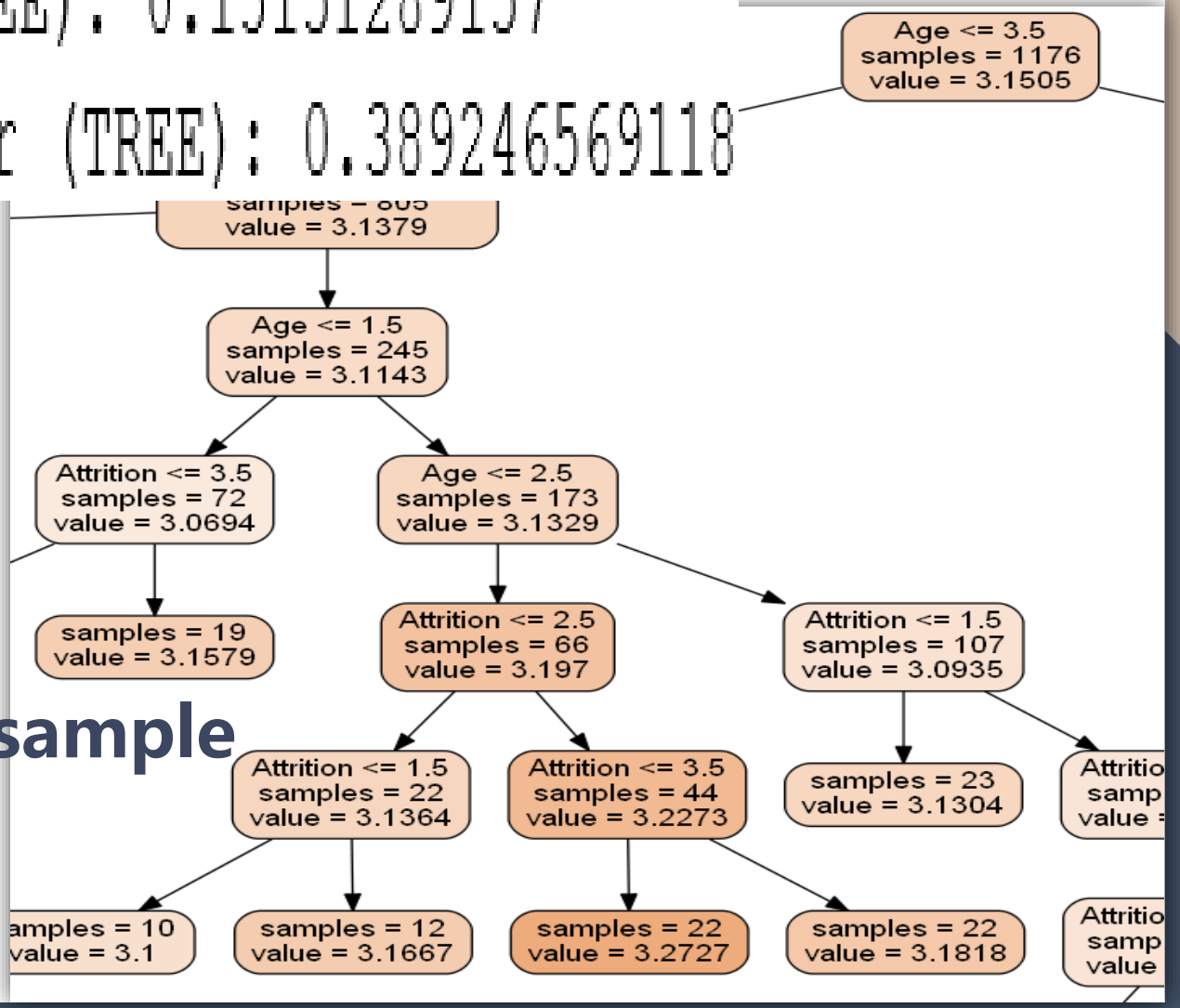
model

Sample :

the number of data

Value :

the value of the corresponding sample



machine learning models (involve all factors)

```
Mean Square Error (FOREST): 0.0
Root Mean Square Error (FOREST): 0.0
The best way is RandomForestRegressor.
```

Assumption: job satisfaction, environment satisfaction and relationship satisfaction are the three most relevant factors to job performance and makes deep research on these three factors.

Despite the fact that using all factors can produce the most accurate results, models based on our assumptions have good results.

Part 5

Summary of the project and future study

After using different methods, especially decision tree, we learn the accuracy of the models of classification to predict employee attrition and job performance more accurately.

Our future study may focus on finding a better models.

When it comes to wide numbers of statistics, our algorithm and models have good results.

Q&A

Thank you!