ĐẠI HỌC QUỐC GIA – THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC KINH TẾ LUẬT

FINAL REPORT

Môn học: DATA MINING

Đề 1: EMPLOYEE LOYALTY

Giảng viên: Ths. NCS. Phan Huy Tâm

Sinh viên thực hiện: Nguyễn Văn Thành

Mã số sinh viên: K204140640

Hồ Chí Minh, ngày 04 tháng 1 năm 2023

I. Khái quát chung về bộ dữ liệu "Employee Loyalty"

Bộ dữ liệu "Employee Loyalty" là dữ liệu sơ cấp, cung cấp thông tin được thu thập về các nhân viên trong một công ty bao gồm hai tập dữ liệu là "employee _satisfaction_ evaluation" và "Hr_data".

1.1. Tổng quan Hr_data

Bộ dữ liệu "Hr_data" có 14999 dòng và 9 cột quan sát, trong đó bao gồm:

Cột thứ nhất "Employee_id": Mã ID của nhân viên.

Cột thứ hai "Number_project": Tổng số dự án đã tham gia của nhân viên.

Cột thứ ba "Avergage_montly_hours": Tổng số giờ làm việc của nhân viên theo tháng.

Cột thứ tư "Time_spend_company": Số năm nhân viên đã làm việc cho công ty.

Cột thứ năm "Worl_accident": Tổng số lần gặp phải tai nạn nghề nghiệp của nhân viên.

Cột thứ sáu "Left": Nhân viên đó có nghỉ việc hay không, 1: nghỉ việc, 2: không nghỉ việc

Cột thứ bảy "Promotion_last_5years": Trong vòng 5 năm trở lại thì nhân viên có thăng tiến hay không, 1: thăng tiến, 2: không thăng tiến.

Cột thứ tám "Department": Vị trí làm việc của nhân viên.

Cột thứ chín "Salary": Mức lương của nhân viên.

***Hình 1.1**: 5 dòng trên cùng*

	employee_id	number_project	average_montly_hours	time_spend_company	Work_accident	left	promotion_last_5years	department	salary
0	1003	2	157	3	0	1	0	sales	low
1	1005	5	262	6	0	1	0	sales	medium
2	1486	7	272	4	0	1	0	sales	medium
3	1038	5	223	5	0	1	0	sales	low
4	1057	2	159	3	0	1	0	sales	low

***Hình 1.2:** 5 dòng dưới cùng*

	employee_id	number_project	average_montly_hours	time_spend_company	Work_accident	left	promotion_last_5years	department	salary
14994	87670	2	151	3	0	1	0	support	low
14995	87673	2	160	3	0	1	0	support	low
14996	87679	2	143	3	0	1	0	support	low
14997	87681	6	280	4	0	1	0	support	low
14998	87684	2	158	3	0	1	0	support	low

Bộ dữ liệu “Hr_data có 2 dữ cột quan sát là dữ liệu phân loại là “department” và “salary”, các cột quan sát còn lại mang dạng dữ liệu “int64”.

Hình 1.3: *Các quan sát trong cột dữ liệu phân loại*

```
check_3 = Hr_data['department'].unique()
check_4 = Hr_data['salary'].unique()
print(f'Cột quan sát department bao gồm  các phòng bang {check_3}')
print(f'Cột quan sát salary bao gồm  các mức lương {check_4}')
```

✓ 0.4s

```
Cột quan sát department bao gồm  các phòng bang ['sales' 'accounting' 'hr' 'technical' 'support' 'management' 'IT
 'product_mng' 'marketing' 'RandD']
Cột quan sát salary bao gồm  các mức lương ['low' 'medium' 'high']
```

1.2. Tổng quan “employee_satisfaction_evaluation”

Bộ dữ liệu “employee_satisfaction_evaluation” có 14999 dòng và 3 cột quan sát, trong đó bao gồm:

Thứ nhất “employee #”: Mã ID của nhân viên.

Thứ hai “satisfaction_level”: Mức độ hài lòng của nhân viên đối với doanh nghiệp, chỉ số trung bình là 0.613.

Thứ ba “last_evaluation”: Lần đánh giá mức độ hài lòng của nhân viên gần đây nhất, chỉ số trung bình là 0.716.

Hình 1.4*: 5 dòng trên cùng*

	EMPLOYEE #	satisfaction_level	last_evaluation
0	1003	0.38	0.53
1	1005	0.80	0.86
2	1486	0.11	0.88
3	1038	0.72	0.87
4	1057	0.37	0.52

***Hình 1.5**: 5 dòng dưới cùng*

	EMPLOYEE #	satisfaction_level	last_evaluation
14994	87670	0.40	0.57
14995	87673	0.37	0.48
14996	87679	0.37	0.53
14997	87681	0.11	0.96
14998	87684	0.37	0.52

1.3. Mục tiêu thực hiện trong bài báo cáo

Hiện nay, mọi công ty đều muốn nắm bắt được tình hình của các nhân viên trong công ty của mình để có thể biết được nhân viên có nhu cầu làm việc tiếp hay không, qua đó có thể dự trù được nhân lực và đào tạo các nhân viên có nhu cầu làm việc gắn bó lâu dài.

Vì vậy, mục tiêu của bài báo cáo là sử dụng mô hình phù hợp để thực hiện dự báo với các yếu tố được cung cấp trên thì nhân viên có tiếp tục làm việc cho công ty hay không.

II. Khám phá dữ liệu

2.1. Tiền xử lý dữ liệu

2.1.1. Gộp hai bộ dữ liệu thành một

Bộ dữ liệu sau khi gộp thì có dữ liệu của cả hai tập "Hr_data" "employee_ satisfaction_evaluation" với 14999 dòng và 11 cột quan sát.

***Hình 2.1:** Code gộp dữ liệu*

```
df = Hr_data.set_index('employee_id').join(employee.set_index('EMPLOYEE #'))
df = df.reset_index()
df.head()
```

2.1.2. Kiểm tra dữ liệu

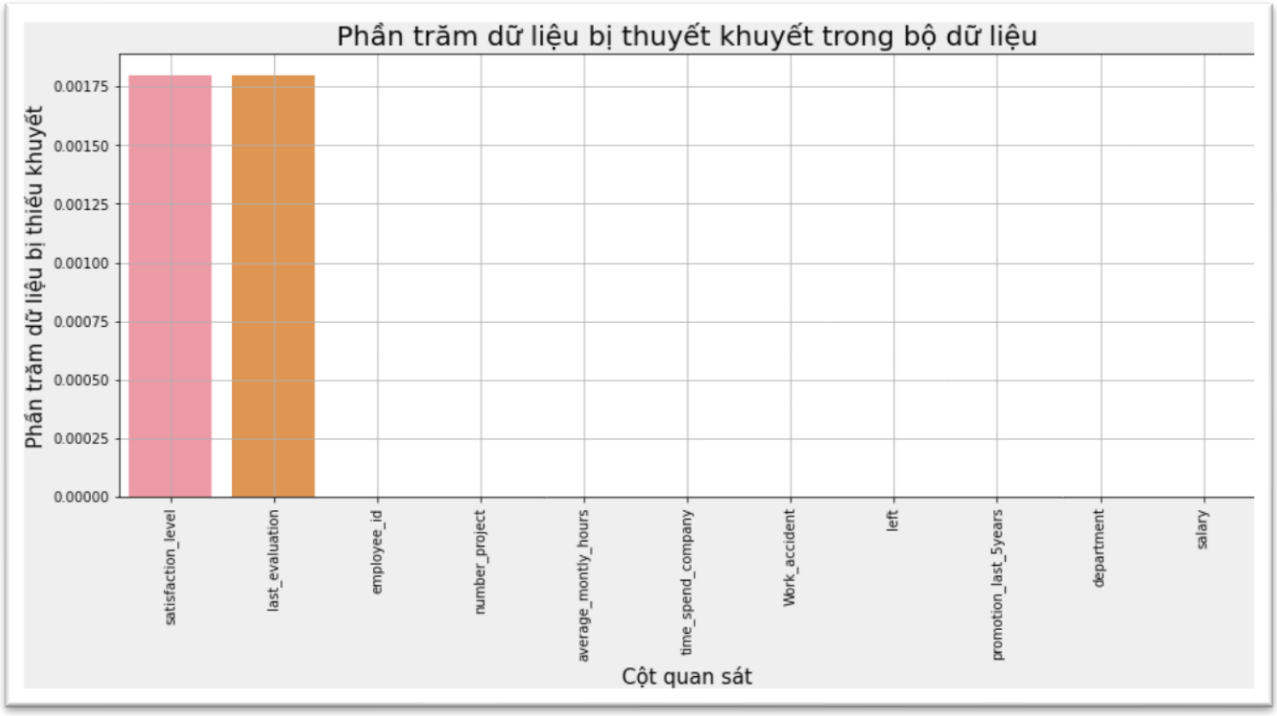
Dữ liệu không có quan sát bị trùng lặp.

Dữ liệu sau khi kiểm tra cho thấy rằng có 54 quan sát bị thiếu, trong đó có 27 quan sát của "satisfaction_level" và "last_evaluation". Dữ liệu khuyết chiếm 0.0018% trong toàn bộ quan sát của mỗi cột chứa dữ liệu khuyết.

***Hình 2.2**: Kết quả kiểm tra dữ liệu thiếu khuyết*

	Total	Percent
satisfaction_level	27	0.0018
last_evaluation	27	0.0018
employee_id	0	0.0000
number_project	0	0.0000
average_montly_hours	0	0.0000
time_spend_company	0	0.0000
Work_accident	0	0.0000
left	0	0.0000
promotion_last_5years	0	0.0000
department	0	0.0000
salary	0	0.0000

***Biểu đồ 2.1**: Phần trăm dữ liệu bị thiếu khuyết*



2.1.3. Xử lý dữ liệu bị khuyết

Dựa trên kết quả sau khi kiểm tra, có thể thấy rằng dữ liệu khuyết chiếm phần trăm rất nhỏ và nằm ở hai cột quan sát *"satisfaction_level" và "last_evaluation"*.

Hai cột quan sát trên đều là quan sát mang dạng định lượng, thang đo tỉ lệ vì vậy có thể giải quyết dữ liệu khuyết theo hướng như sau:

Bước 1: Loại bỏ các dòng quan sát có 2 dữ liệu khuyết.

Bước 2: Điền chỉ số trung bình của cột quan sát vào dữ liệu bị khuyết.

Giải thích: Với dữ liệu dạng định lượng thì việc sử dụng các giá trị trung bình hoặc trung vị là một lựa chọn an toàn vì các giá trị đó trong thực tế có khả năng cao sẽ xảy ra.

Hình 2.3: *Code và phần trăm dữ liệu còn lại sau khi xử lý*

```
Delete_per = (df.shape[0]/df_check.shape[0])*100
print(f"Phần trăm còn lại của dữ liệu là {Delete_per}")
```

✓ 0.4s

```
Phần trăm còn lại của dữ liệu là 99.92666177745183
```

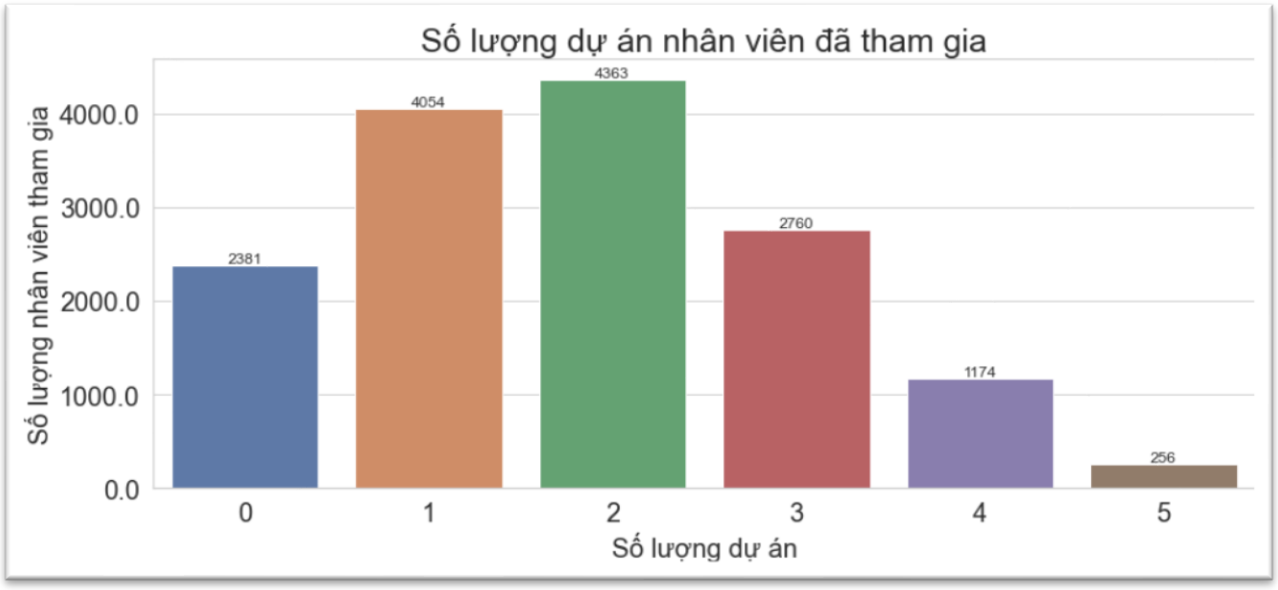
Có thể thấy dữ liệu bị xóa đi là khoảng 0,08%.

2.2. Trực quan hóa dữ liệu

2.2.1. Các quan sát định lượng

Thứ nhất: Cột quan sát Number_project

Biểu đồ 2.2: *Số lượng dự án nhân viên đã tham gia*



Nhận xét biểu đồ 2.2: có thể thấy rằng nhân viên đã tham gia 4 dự án là nhiều nhất với 4363 nhân viên, sau đó là 3 dự án với 4054 nhân viên và thấp nhất là 7 dự án 256 nhân viên.

Quyết định: Kiểm tra thử các nhân viên tham gia 7 dự án có thăng chức hay không, họ có nghỉ việc hay không.

Hình 2.4: *Kết quả kiểm tra nhân viên tham gia 7 dự án*

```
df_test = df[(df['number_project'] == 7)][['promotion_last_5years','left']]
df_test.value_counts()
```

✓ 0.9s

```
promotion_last_5years  left
0                      1       256
dtype: int64
```

Nhận xét hình 2.4: Có thể thấy rằng các nhân viên tham gia 7 dự án nhưng không có sự thăng tiến trong 5 năm qua đều quyết định nghỉ việc.

Quyết định: Kiểm tra thử các nhân viên tham gia 6 dự án có thăng chức hay không, họ có nghỉ việc hay không.

Hình 2.5: *Kết quả kiểm tra nhân viên tham gia 6 dự án*

```
df_test_1 = df[(df['number_project'] == 6)][['promotion_last_5years','left']]
df_test_1.value_counts()
```

✓ 0.6s

```
promotion_last_5years  left
0                      1       652
                       0       502
1                      0        17
                       1         3
dtype: int64
```

Nhận xét hình 2.5: 625 nhân viên làm 6 dự án quyết định nghỉ khi không có sự thăng tiến trong vòng 5 năm qua quyết định nghỉ việc và 502 nhân viên tiếp tục làm việc.

Có thể thấy rằng các nhân viên khi tham gia nhiều dự án nhưng không được thăng tiến đều có xu hướng nghỉ việc tại công ty.

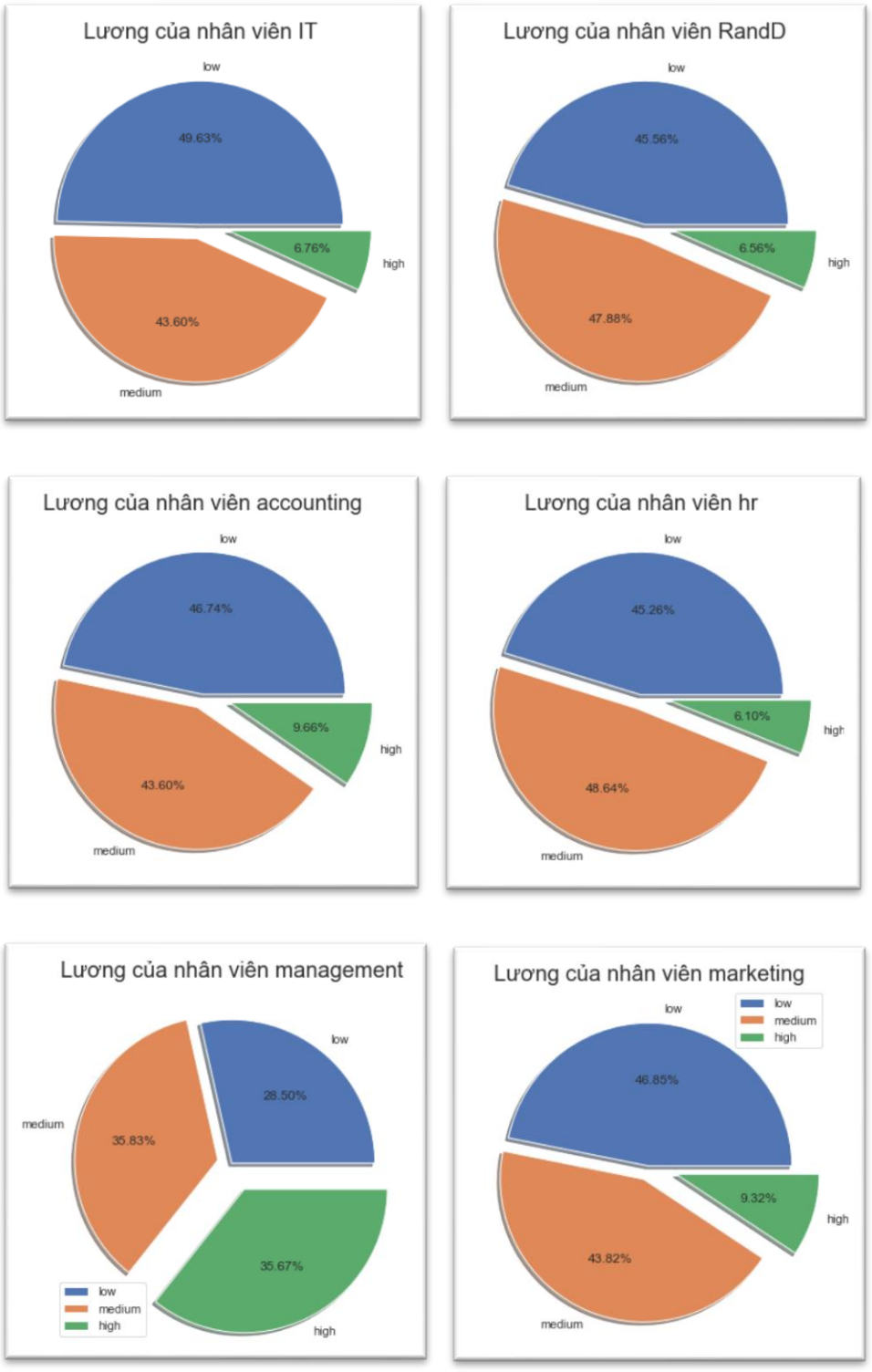
Thứ hai: Cột quan sát mục tiêu dự báo "left"

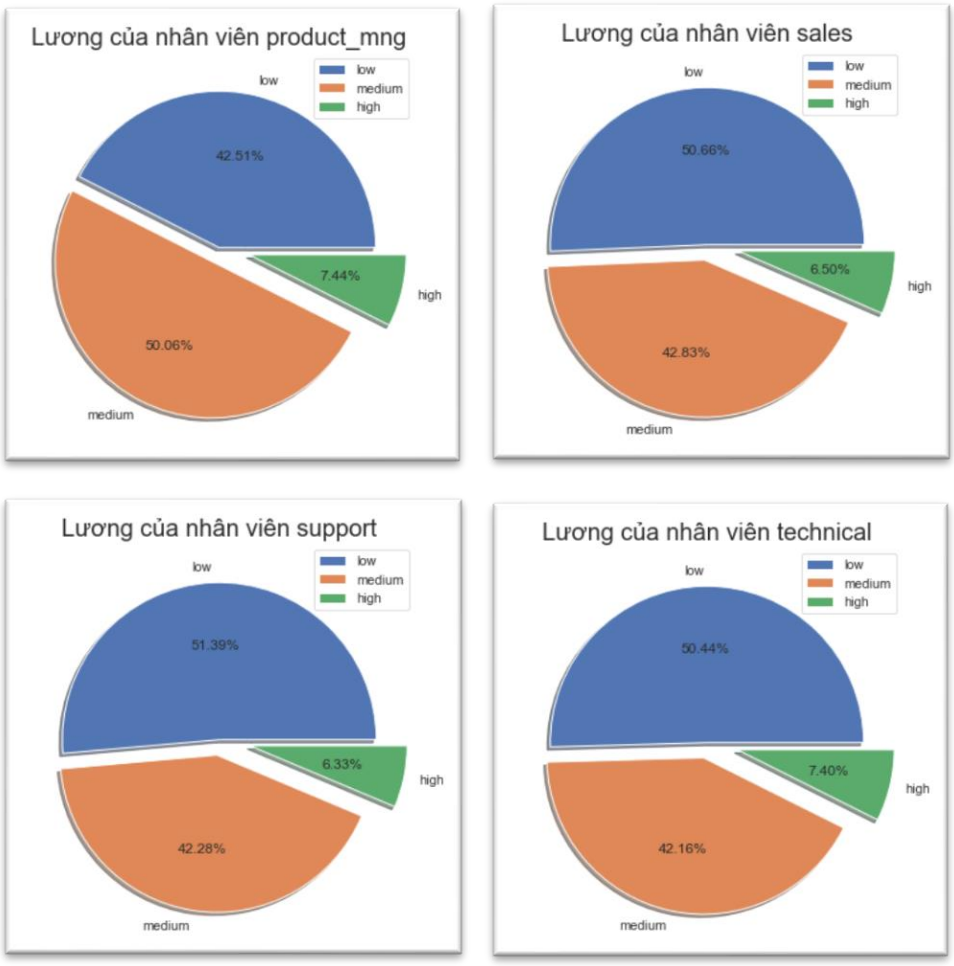
Biểu đồ 2.3: *So sánh số lượng nhân viên nghỉ việc và không nghỉ việc*

Nhận xét biểu đồ 2.3: Có thể thấy rằng số lượng nhân viên nghỉ việc chiếm 23,77%, thấp hơn hẳn so với nhân viên vẫn tiếp tục đi làm.

2.2.2. Các quan sát định tính "department" và "salary"

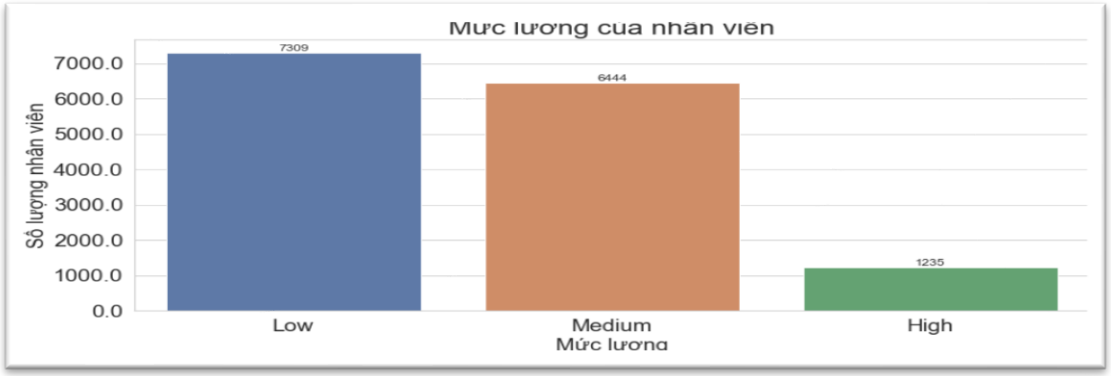
Biểu đồ 2.4: So sánh mức lương từng phòng ban





Nhận xét biểu đồ 2.3: Có thể thấy rằng phần lớn mức lương của các nhân viên đều nằm ở mức thấp và trung bình. Chỉ có mức lương của nhân viên management cân bằng ở cả ba mức lương.

Biểu đồ 2.5: *Mức lương của nhân viên*



Quyết định: Kiểm tra xem các nhân viên có mức lương thấp và không có sự thăng tiến trong vòng 5 năm qua có quyết định nghỉ việc hay không.

Hình 2.6: *Kết quả Kiểm tra nhân viên mức lương thấp và không thăng tiến*

```
df_test_2 = df[(df['salary'] == 'low')][['promotion_last_5years','left']]
df_test_2.value_counts()
```

✓ 0.1s

```
promotion_last_5years  left
0                      0       5092
                       1       2152
1                      0         52
                       1         13
dtype: int64
```

Nhận xét hình 2.6: Phần lớn các nhân viên có mức lương thấp và không có sự thăng tiến đều quyết định nghỉ việc.

Quyết định: Kiểm tra xem các nhân viên có mức lương cao và không có sự thăng tiến trong vòng 5 năm qua có quyết định nghỉ việc hay không.

Hình 2.7: Kết quả kiểm tra nhân viên có mức lương cao và không thăng tiến

```
df_test_2 = df[(df['salary'] == 'high')][['promotion_last_5years','left']]
df_test_2.value_counts()
```

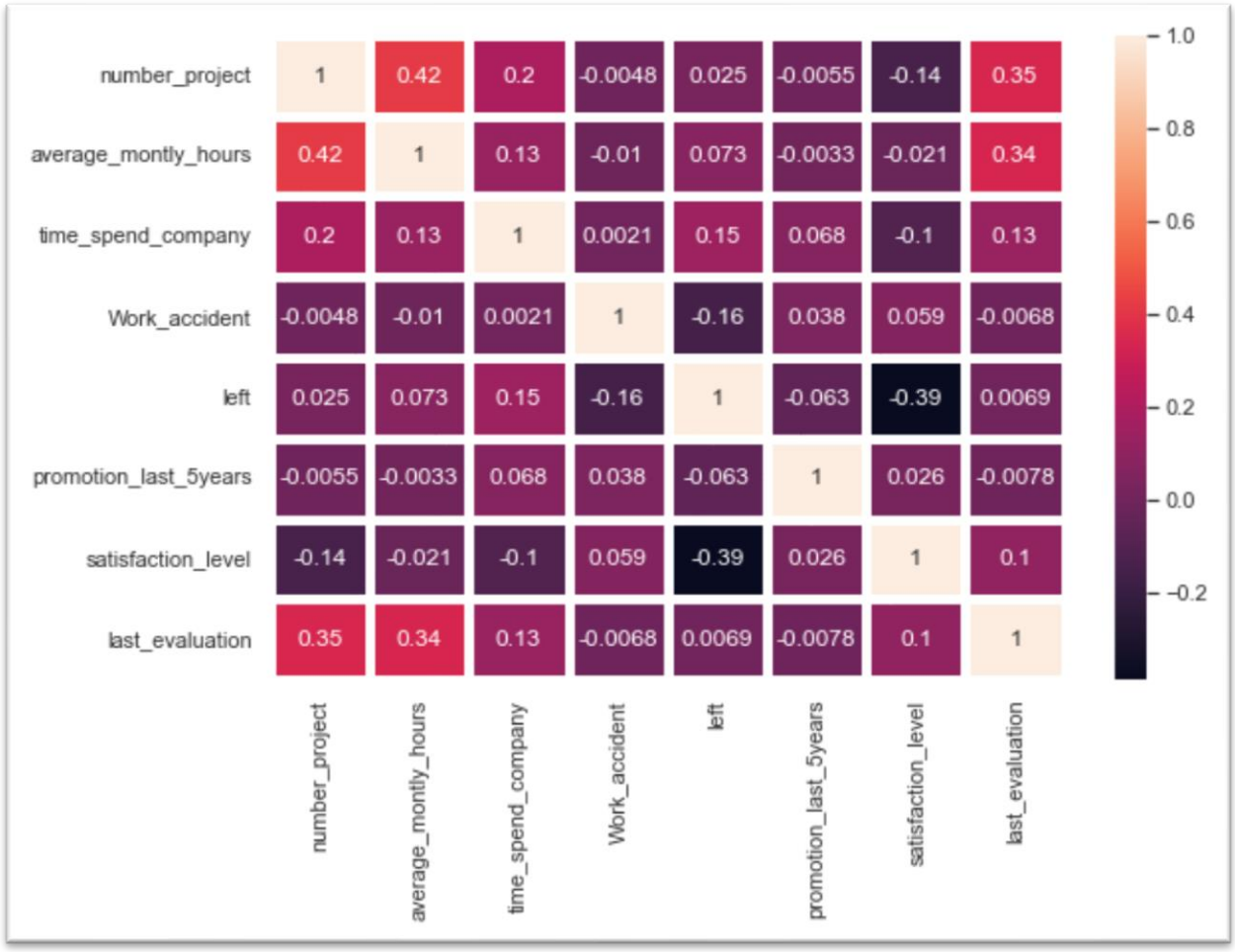
✓ 0.9s

```
promotion_last_5years  left
0                      0       1082
                       1         81
1                      0         72
dtype: int64
```

Nhận xét hình 2.7: Phần lớn các nhân viên có mức lương cao nhưng không có sự thăng tiến trong vòng 5 năm qua đều quyết định nghỉ việc.

2.2.3. Mức độ tương quan giữa các quan sát

Biểu đồ 2.6: *Mức độ tương quan giữa các quan sát*



Nhận xét biểu đồ 2.6:

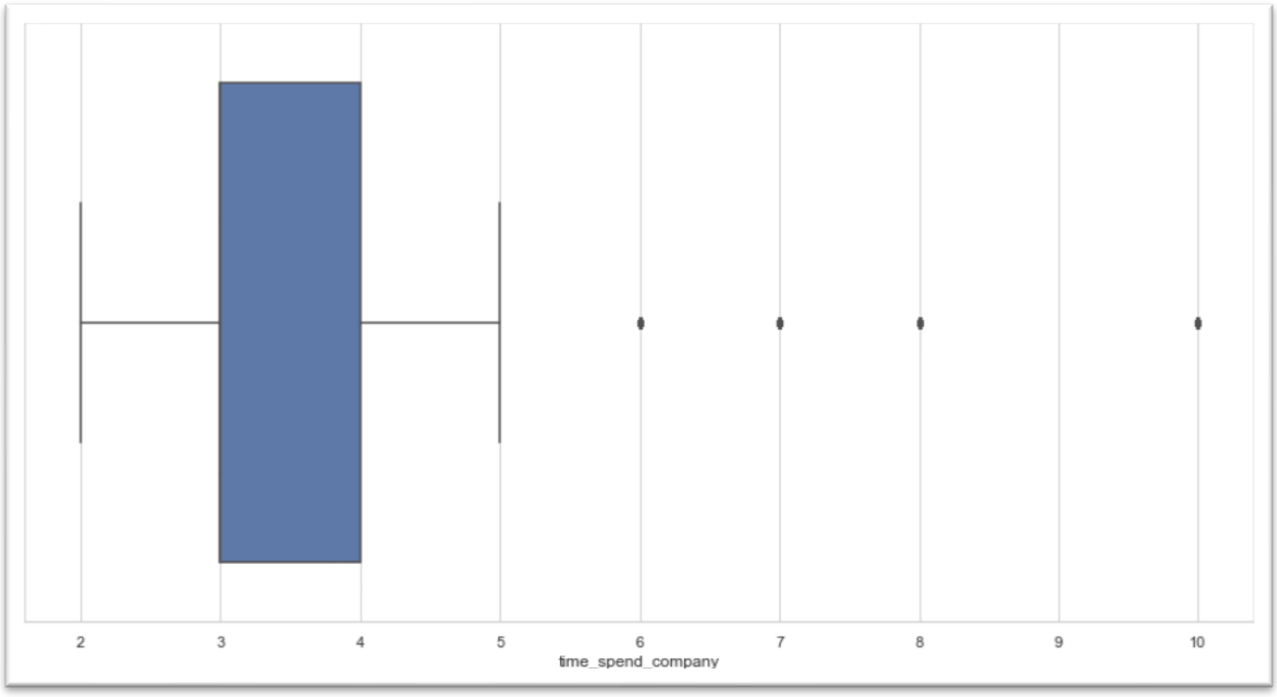
Có thể thấy rằng quan sát *"left"* tương quan thuận và yếu với đánh giá lần cuối của nhân viên, thời gian làm việc của nhân viên dành cho công ty và số lượng dự án đã tham gia. Hệ số tương quan với các quan sát này nằm trong khoảng từ 0,0069 đến 0,15.

Quyết định bỏ việc của công ty tương quan nghịch với các quan sát là đánh giá mức độ hài lòng của nhân viên với mức tương quan là trung bình, sự thăng tiến và tai nạn nghề nghiệp với mức tương quan yếu.

2.2.4. Xử lý dữ liệu ngoại lai

Kiểm tra dữ liệu ngoại lai: Kết quả sau khi kiểm tra dữ liệu ngoại lai thì chỉ có cột quan sát "time_spend_caompany" là xuất hiện giá trị ngoại lai.

Biểu đồ 2.7: *Boxplot "time_spend_caompany"*



Quyết định: Xử lý giá trị ngoại lai bằng phương pháp IQR.

IQR (viết tắt của "interquartile range") là khoảng trải nghiệm giữa hay còn được gọi là khoảng tứ phân vị của tệp dữ liệu. Phương pháp này sẽ loại bỏ các giá trị dưới Q1 ít nhất là 1,5*IQR hoặc nằm trên Q3 ít nhất là 1,5*IQR.

Hình 2.8: *Kết quả sau khi xử lý ngoại lai*

```
Q1 = df['number_project'].quantile(0.25)
Q3 = df['number_project'].quantile(0.75)
IQR = Q3 - Q1
print(f'Chỉ số IQR là: {IQR} ')

Test_1 = ~((df['time_spend_company'] > (Q3 + 1.5 * IQR)))
Subequal = df[Test_1]
Test_2 = ~((df['time_spend_company'] < (Q3 + 1.5 * IQR)))
Outstanding = df[Test_2]
print(f'Dữ liệu còn lại sau khi lấy outlier là {(Subequal.shape[0] / df.shape[0]) * 100} %')
print(Subequal.shape)
Subequal.head()
```

✓ 0.5s

```
Chỉ số IQR là: 2.0
Dữ liệu còn lại sau khi lấy outlier là 98.5721910862023 %
(14774, 10)
```

2.2.5. Đánh giá chung

Bộ dữ liệu *"employee loyalty"* không có nhiều dữ liệu thiếu khuyết và các giá trị ngoại lai không nhiều, mức độ tương quan giữa các cột quan sát từ yếu đến trung bình yếu.

Mức lương phần lớn của các nhân viên trong công ty là thấp và trung bình. Các nhân viên có mức lương cao, nhân viên tham gia nhiều dự án có xu hướng nghỉ việc khi không được thăng tiến trong vòng 5 năm qua.

Quan sát mục tiêu *"left"* có sự chênh lệch lớn giữa nhân viên nghỉ việc và không nghỉ việc, vì vậy cần chú ý khi chạy mô hình cần phải chia tập dữ liệu đều giữa tập train và test.

III. Dự báo với Machine Learning

Các thư viện cần dùng cho phần dự báo với Machine Learning:

from sklearn.model_selection import train_test_split

3.1. Chuẩn bị dữ liệu cho mô hình

Bước 1: Chuẩn hóa dữ liệu với phương pháp one-hot encoding.

Trong bộ dữ liệu có 2 biến quan sát thuộc dạng định tính, vì vậy cần phải chuẩn hóa về dạng số để có thể chạy mô hình.

Hình 3.1: *Kết quả sau khi chuẩn hóa*

```
categorial = ['department','salary']
df_ML = pd.get_dummies(df_ML, columns=categorial, drop_first=True)
shape1 = df_ML.shape[1] - df.shape[1]
print(f'Dữ liệu sau khi chuẩn hóa có thêm {shape1} cột')
```

✓ 0.6s

```
Dữ liệu sau khi chuẩn hóa có thêm 9 cột
```

Bước 2: Loại bỏ biến mục tiêu "left" ra khỏi bộ dữ liệu và gán vào X,y.

Hình 3.2: *Code thực hiện bước 2*

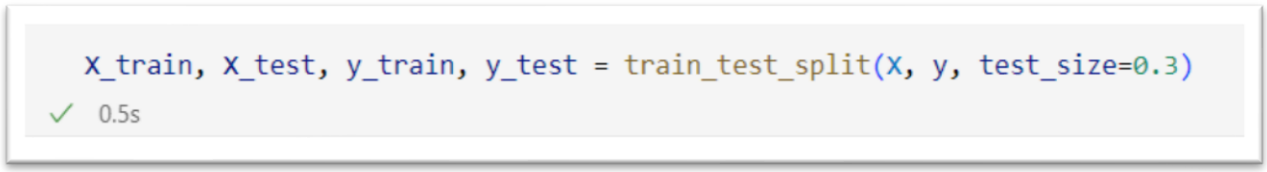
```
# Loại bỏ biến dự báo, chỉ định dữ liệu còn lại vào X
X = df_ML.drop(['left'],axis=1).values

# Chỉ định biến dự báo vào y
y = df_ML['left'].values
```

Bước 3: Chia tập dữ liệu

Chia tập dữ liệu thành 2 phần là phần test và train với tỷ lệ là 70:30.

***Hình 3.3**: Code thực hiện bước 3*



Bước 4: Chuẩn hóa lại bộ dữ liệu mới phương pháp

Các giá trị trong dữ liệu không dùng chung đơn vị tính, vì vậy cần chuẩn hóa, quy đổi các tỷ lệ để có thể so sánh.

***Hình 3.4**: Code thực hiện bước 4*

3.2. Lựa chọn mô hình dự báo

3.2.1. Mô hình Linear Regression

Mô hình thuật toán Linear Regression là một trong những thuật toán cơ bản nhất của Machine Learning, thuộc nhóm Supervised Learning (học có giám sát). Tuy là thuật toán cơ bản nhưng nó đã được chứng minh được tính hữu ích cho một lượng lớn các tình huống. Đây là một kỹ thuật thống kê đã được sử dụng từ lâu, nhiều phương pháp khoa học và trí tuệ nhân tạo đều có thể sử dụng hồi quy để giải quyết những bài toán phức tạp

3.2.2. Mô hình hồi quy Logictic

Hồi quy logistic là một kỹ thuật phân tích dữ liệu sử dụng toán học để tìm ra mối quan hệ giữa hai yếu tố dữ liệu. Sau đó, kỹ thuật này sử dụng mối quan hệ đã được tìm thấy để dự đoán giá trị của những yếu tố cơ bản dựa trên yếu tố còn lại. Dự đoán thường cho ra một số kết quả hữu hạn, như có hoặc không**.**

Hồi quy logistic có thể dùng để thực hiện các tác vụ xử lý dữ liệu phức tạp để dự đoán và quản lý. Một số lợi ích của việc sử dụng hồi quy logictic so với các kỹ thuật máy học khác như: Tính đơn giản, tốc độ xử lý, sự linh hoạt và khả năng hiển thị.

3.2.3. Mô hình K-Nearest Neighbors

Thuật toán K lân cận gần nhất trong tiếng Anh là K-Nearest Neighbor, viết tắt là KNN, một kỹ thuật học thuật có giám sát (học có giám sát) được sử dụng để phân loại quan sát mới bằng cách tìm điểm tương đồng giữa quan sát mới này với dữ liệu có sẵn. Khi đào tạo, thuật toán này không học một điều gì từ đào tạo dữ liệu (đây cũng là lý do thuật toán này được xếp vào loại lười học), mọi tính toán đều được thực hiện khi nó cần dự đoán kết quả của dữ liệu mới. K-gần nhất hàng xóm có thể áp dụng được vào cả hai loại bài toán Học có giám sát là Phân loại và Hồi quy.

3.2.4. Mô hình Random Forests

Mặc dù mô hình Random Forests đưa ra dự báo chậm bởi vì có tích hợp nhiều cây quyết định. Tuy nhiên, Random Forests được coi là một phương pháp chính xác và mạnh mẽ vì số cây quyết định tham gia vào quá trình này. Nó không phải là vấn đề Overfitting. Lý do chính là nó làm mất trung bình của tất cả các dự đoán, trong đó hủy bỏ những thành kiến. Thuật toán có thể được sử dụng trong cả hai vấn đề phân loại và quy định hồi quy.

3.3. Chạy mô hình Machine Learning

3.3.1. Mô hình Linear Regression

Hình 3.5: *Kết quả dự báo LR*

Kết quả dự báo đúng 79.05%

```
[[3198  256]
 [ 686  357]]
```

	precision	recall	f1-score	support
0	0.82	0.93	0.87	3454
1	0.58	0.34	0.43	1043
accuracy			0.79	4497
macro avg	0.70	0.63	0.65	4497
weighted avg	0.77	0.79	0.77	4497

Nhận xét kết quả thuật toán LR: Kết quả dự báo của thuật toán LR là 79,05%.

Dự báo đúng không nghỉ việc là 3198 nhân viên , không nghỉ việc nhưng thực tế nghỉ việc là 256 nhân viên.

Dự báo đúng nghỉ việc là 686 nhân viên, nhân viên nghỉ việc nhưng trên thực tế không nghỉ việc là 357 nhân viên.

Precision là 0,82 cho nhân viên không nghỉ việc và 0,58 cho nhân viên nghỉ việc.

Recall là 0,93 cho nhân viên không nghỉ việc và 0,34 cho nhân viên nghỉ việc.

3.3.2. *Mô hình hồi quy Logictic*

Hình 3.6: *Kết quả hồi quy Logictic*

```
Kết quả dự báo đúng 67.38%
[[2282 1172]
 [ 295  748]]
              precision    recall  f1-score   support

           0       0.89      0.66      0.76      3454
           1       0.39      0.72      0.50      1043

    accuracy                           0.67      4497
   macro avg       0.64      0.69      0.63      4497
weighted avg       0.77      0.67      0.70      4497
```

Nhận xét kết quả thuật toán Logictic: Kết quả dự báo của thuật toán *Logictic* là 67,38%

Dự báo đúng không nghỉ việc là 2282 nhân viên , không nghỉ việc nhưng thực tế nghỉ việc là 1172 nhân viên.

Dự báo đúng nghỉ việc là 295 nhân viên, nhân viên nghỉ việc nhưng trên thực tế không nghỉ việc là 748 nhân viên.

Precision là 0,89 cho nhân viên không nghỉ việc và 0,39 cho nhân viên nghỉ việc.

Recall là 0,66 cho nhân viên không nghỉ việc và 0,72 cho nhân viên nghỉ việc.

3.3.3. Mô hình K-Nearest Neighbors

Hình 3.7: *Kết quả dự báo KNN*

```
Kết quả dự báo đúng 96.60%
[[3343  111]
 [  42 1001]]
              precision    recall  f1-score   support

           0       0.99      0.97      0.98      3454
           1       0.90      0.96      0.93      1043

    accuracy                           0.97      4497
   macro avg       0.94      0.96      0.95      4497
weighted avg       0.97      0.97      0.97      4497
```

Nhận xét kết quả thuật toán KNN: Kết quả dự báo của thuật toán *KNN* là 96,6%

Dự báo đúng không nghỉ việc là 3343 nhân viên , không nghỉ việc nhưng thực tế nghỉ việc là 111 nhân viên.

Dự báo đúng nghỉ việc là 42 nhân viên, nhân viên nghỉ việc nhưng trên thực tế không nghỉ việc là 1001 nhân viên.

Precision là 0,99 cho nhân viên không nghỉ việc và 0,9 cho nhân viên nghỉ việc.

Recall là 0,97 cho nhân viên không nghỉ việc và 0,96 cho nhân viên nghỉ việc.

3.3.4. Mô hình Random Forests

Hình 3.8: *Kết quả dự báo RF*

```
Accuracy 99.09%
[[3444   10]
 [  31 1012]]
              precision    recall  f1-score   support

           0       0.99      1.00      0.99      3454
           1       0.99      0.97      0.98      1043

    accuracy                           0.99      4497
   macro avg       0.99      0.98      0.99      4497
weighted avg       0.99      0.99      0.99      4497
```

Nhận xét kết quả thuật toán RF: Kết quả dự báo của thuật toán *RF* là 99,09 %

Dự báo đúng không nghỉ việc là 3444 nhân viên , không nghỉ việc nhưng thực tế nghỉ việc là 10 nhân viên.

Dự báo đúng nghỉ việc là 31 nhân viên, nhân viên nghỉ việc nhưng trên thực tế không nghỉ việc là 1012 nhân viên.

Precision là 0,99 cho nhân viên không nghỉ việc và 0,99 cho nhân viên nghỉ việc.

Recall là 1,0 cho nhân viên không nghỉ việc và 0,96 cho nhân viên nghỉ việc.

3.4. Đánh giá mô hình

Các tiêu chí để chọn mô hình phù hợp như sau:

Thứ nhất: Tỷ lệ dự báo chính xác cao.

Thứ hai: Tỷ lệ sai lầm loại 1 và 2 thấp. Khi công ty đào tạo nhân viên, thường sẽ muốn tập trung vào các nhân viên có năng lực và gắn bó với công ty lâu dài. Vì vậy khi chọn nhầm nhân viên để làm trọng tâm bồi dưỡng nhưng họ quyết định nghỉ việc sẽ mang lại tổn thất lớn cho công ty. Ngoài ra, các nhân viên có năng lực nhưng không thể được bồi dưỡng và thăng cấp thì công ty sẽ mất đi nhân viên có năng lực.

Như vậy, dựa vào kết quả các mô hình trên, có thể thấy rằng 2 mô hình với thuật toán KNN và Random Forest mang lại kết quả dự báo cao với 96,6% và 99,09%.

Với thuật toán KNN: Phần trăm sai lầm loại 1 của thuật toán KNN là 3,21%, phần trăm sai lầm loại 2 là: 95,9%.

Với thuật toán RF: Phần trăm sai lầm loại 1 của thuật toán KNN là 0,003 %, phần trăm sai lầm loại 2 là: 97,02%.

Kết luận: Mô hình dự báo với thuật toán Random Forest mang lại hiệu quả cao với mức dự báo đúng lên đến 99,09% và sai lầm loại 1 nằm ở mức 0,003%. Mặc dù sai lầm loại 2 cao đến 97,02%, cao hơn sai lầm loại 2 của thuật toán KNN. Tuy nhiên công ty khi quyết định bồi dưỡng nhầm nhân viên thì sẽ tốn nhiều tài chính và nhân sự, khi họ quyết định nghỉ việc sẽ mang lại tổn thất rất lớn, nhân viên có năng lực không được bồi dưỡng khi quyết định nghỉ việc thì công ty chỉ có tổn thất về mặt nhân sự. Vì vậy mô hình dự báo với thuật toán Random Forest phù hợp trong trường hợp này.

IV. Kết luận và khuyến nghị

Dựa vào phân tích trên bộ dữ liệu *"Employee Loyalty"*, nhân viên làm việc tại công ty khi làm việc lâu dài cho công ty, tham gia nhiều dự án nhưng không được thăng tiến thì sẽ có xu hướng nghỉ việc. Các nhân viên có mức thu thập cao cũng sẽ có xu hướng nghỉ việc khi không được thăng chức.

Công ty nên có các chính sách quan tâm về nhân sự, chính sách thăng tiến dành cho các nhân viên có năng lực đã làm việc nhiều năm với công ty để có thể tạo thêm động lực cho nhân viên tiếp tục làm việc và gắn bó lâu dài với công ty. Ngoài ra, công ty cần phải xác định đúng trọng tâm bồi dưỡng nhân viên có năng lực để tránh tổn thất ngoài ý muốn. Công ty nên có những chính sách bồi dưỡng và đạo tạo nhân lực kế thừa để khi các nhân viên cũ quyết định nghỉ việc sẽ có thể bổ sung vào vị trí trống, mang lại hiệu quả cao trong việc phát triển công ty, tránh tình trạng thiếu nhân lực trong quá trình hoạt động.

Link github chứ file code: https://github.com/MiCasa0403001/EMPLOYEE-LOYALTY/blob/main/K204140640.ipynb

Trích nguồn tham khảo

https://luanvanhay.org/dich-vu/top-10-mo-hinh-hoc-may-cho-thong-ke-hoc-thuat/

https://aws.amazon.com/vi/what-is/logistic-regression/

https://aws.amazon.com/vi/what-is/linear-regression/

https://data36.com/random-forest-in-python/

https://machinelearningcoban.com/2017/01/08/knn/

https://vietnambiz.vn/thuat-toan-k-lang-gieng-gan-nhat-k-nearest-neighbor-knn-la-gi-2020022911113334.htm#:~:text=KNN%20l%C3%A0%20m%E1%BB%99t%20m%C3%B4%20h%C3%ACnh,%C4%91%E1%BB%83%20ph%C3%A2n%20lo%E1%BA%A1i%20%C4%91a%20l%E1%BB%9Bp.

https://matplotlib.org/

https://seaborn.pydata.org/index.html

https://machinelearningcoban.com/2017/08/31/evaluation/

https://machinelearningcoban.com/2017/01/27/logisticregression/