Activity_ Course 7 Salifort Motors project lab

July 15, 2023

1 Capstone project: Providing data-driven suggestions for HR

1.1 Description and deliverables

This capstone project is an opportunity for you to analyze a dataset and build predictive models that can provide insights to the Human Resources (HR) department of a large consulting firm.

Upon completion, you will have two artifacts that you would be able to present to future employers. One is a brief one-page summary of this project that you would present to external stakeholders as the data professional in Salifort Motors. The other is a complete code notebook provided here. Please consider your prior course work and select one way to achieve this given project question. Either use a regression model or machine learning model to predict whether or not an employee will leave the company. The exemplar following this actiivty shows both approaches, but you only need to do one.

In your deliverables, you will include the model evaluation (and interpretation if applicable), a data visualization(s) of your choice that is directly related to the question you ask, ethical considerations, and the resources you used to troubleshoot and find answers or solutions.

2 PACE stages

2.1 Pace: Plan

Consider the questions in your PACE Strategy Document to reflect on the Plan stage.

In this stage, consider the following:

2.1.1 Understand the business scenario and problem

The HR department at Salifort Motors wants to take some initiatives to improve employee satisfaction levels at the company. They collected data from employees, but now they don't know what to do with it. They refer to you as a data analytics professional and ask you to provide data-driven suggestions based on your understanding of the data. They have the following question: what's likely to make the employee leave the company?

Your goals in this project are to analyze the data collected by the HR department and to build a model that predicts whether or not an employee will leave the company.

If you can predict employees likely to quit, it might be possible to identify factors that contribute to their leaving. Because it is time-consuming and expensive to find, interview, and hire new employees, increasing employee retention will be beneficial to the company.

2.1.2 Familiarize yourself with the HR dataset

The dataset that you'll be using in this lab contains 15,000 rows and 10 columns for the variables listed below.

Note: you don't need to download any data to complete this lab. For more information about the data, refer to its source on Kaggle.

Variable	Description
satisfaction_level	Employee-reported job satisfaction level [0–1]
last_evaluation	Score of employee's last performance review [0–1]
number_project	Number of projects employee contributes to
average_monthly_hours	Average number of hours employee worked per month
time_spend_company	How long the employee has been with the company (years)
Work_accident	Whether or not the employee experienced an accident while at work
left	Whether or not the employee left the company
promotion_last_5years	Whether or not the employee was promoted in the last 5 years
Department	The employee's department
salary	The employee's salary (U.S. dollars)

Reflect on these questions as you complete the plan stage.

- Who are your stakeholders for this project?
- What are you trying to solve or accomplish?
- What are your initial observations when you explore the data?
- What resources do you find yourself using as you complete this stage? (Make sure to include the links.)
- Do you have any ethical considerations in this stage?

The HR department at Salifort Motors wants to improve employee satisfaction levels and seeks data-driven suggestions based on the collected data. The primary question they have is: What factors contribute to employees leaving the company? Analyze the HR dataset to gain insights and build a predictive model that can accurately predict employee attrition. Identify factors that contribute to employees leaving the company. Provide data-driven recommendations to improve employee retention. The dataset can be accessed from Kaggle. Initial exploratory data analysis and data visualization can help gain insights into the data and potential relationships between variables. Utilize data manipulation packages such as NumPy and Pandas for data handling and analysis. Employ data visualization libraries like Matplotlib and Seaborn for visual exploration. Refer to relevant links and resources for troubleshooting and finding answers to specific questions. Ensure data privacy and confidentiality. Handle sensitive employee information responsibly and securely. Avoid biases in data analysis and model building.

Analyze

2.2 Step 1. Imports

- Import packages
- Load dataset

2.2.1 Import packages

[1]:

```
# Import packages
### YOUR CODE HERE ###

# For data manipulation
import numpy as np
import pandas as pd

# For data visualization
import matplotlib.pyplot as plt
import seaborn as sns
```

2.2.2 Load dataset

`Pandas` is used to read a dataset called **`HR_capstone_dataset.csv`.** As shown in this cell, the dataset has been automatically loaded in for you. You do not need to download the .csv file, or provide more code, in order to access the dataset and proceed with this lab. Please continue with this activity by completing the following instructions.

[3]:

```
# RUN THIS CELL TO IMPORT YOUR DATA.

# Load dataset into a dataframe
### YOUR CODE HERE ###
df0 = pd.read_csv("HR_capstone_dataset.csv")


# Display first few rows of the dataframe
### YOUR CODE HERE ###
df0.head()
```

```
[3]:    satisfaction_level  last_evaluation  number_project  average_montly_hours  \
     0                0.38             0.53               2                   157
     1                0.80             0.86               5                   262
     2                0.11             0.88               7                   272
     3                0.72             0.87               5                   223
     4                0.37             0.52               2                   159

        time_spend_company  Work_accident  left  promotion_last_5years Department  \
     0                   3              0     1                      0      sales
     1                   6              0     1                      0      sales
     2                   4              0     1                      0      sales
     3                   5              0     1                      0      sales
     4                   3              0     1                      0      sales

        salary
     0     low
     1  medium
     2  medium
     3     low
     4     low
```

2.3 Step 2. Data Exploration (Initial EDA and data cleaning)

- Understand your variables
- Clean your dataset (missing data, redundant data, outliers)

2.3.1 Gather basic information about the data

```
[5]: # Gather basic information about the data
     ### YOUR CODE HERE ###
     print("Dataset dimensions:", df0.shape)
```

```
Dataset dimensions: (14999, 10)
```

Gather descriptive statistics about the data

```
[7]: # Gather descriptive statistics about the data
     ### YOUR CODE HERE ###
     print(df0.head())
     print(df0.describe())
```

```
   satisfaction_level  last_evaluation  number_project  average_montly_hours  \
0                0.38             0.53               2                   157
1                0.80             0.86               5                   262
2                0.11             0.88               7                   272
3                0.72             0.87               5                   223
```

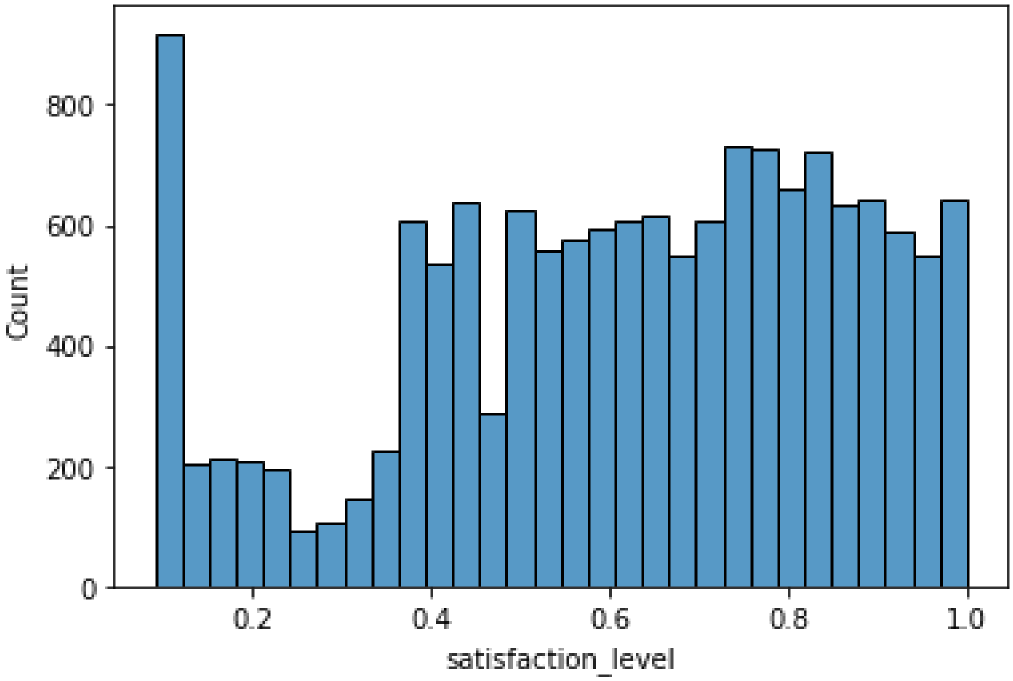
```
4                0.37             0.52               2                  159

   time_spend_company  Work_accident  left  promotion_last_5years Department  \
0                   3              0     1                      0      sales
1                   6              0     1                      0      sales
2                   4              0     1                      0      sales
3                   5              0     1                      0      sales
4                   3              0     1                      0      sales

   salary
0     low
1  medium
2  medium
3     low
4     low
       satisfaction_level  last_evaluation  number_project  \
count        14999.000000     14999.000000    14999.000000
mean             0.612834         0.716102        3.803054
std              0.248631         0.171169        1.232592
min              0.090000         0.360000        2.000000
25%              0.440000         0.560000        3.000000
50%              0.640000         0.720000        4.000000
75%              0.820000         0.870000        5.000000
max              1.000000         1.000000        7.000000

       average_montly_hours  time_spend_company  Work_accident          left  \
count          14999.000000        14999.000000   14999.000000  14999.000000
mean             201.050337            3.498233       0.144610      0.238083
std               49.943099            1.460136       0.351719      0.425924
min               96.000000            2.000000       0.000000      0.000000
25%              156.000000            3.000000       0.000000      0.000000
50%              200.000000            3.000000       0.000000      0.000000
75%              245.000000            4.000000       0.000000      0.000000
max              310.000000           10.000000       1.000000      1.000000

       promotion_last_5years
count           14999.000000
mean                0.021268
std                 0.144281
min                 0.000000
25%                 0.000000
50%                 0.000000
75%                 0.000000
max                 1.000000
```

Examine the data types of each column,check for missing values and analyze the distribution and relationships between variables

[15]:

```python
print(df0.isnull().sum())
print(df0.dtypes)
sns.histplot(data=df0, x='satisfaction_level')
plt.show()


correlation = df0.corr()
print(correlation)
```

```
satisfaction_level       0
last_evaluation          0
number_project           0
average_montly_hours     0
time_spend_company       0
Work_accident            0
left                     0
promotion_last_5years    0
Department               0
salary                   0
dtype: int64
satisfaction_level       float64
last_evaluation          float64
number_project             int64
average_montly_hours       int64
time_spend_company         int64
Work_accident              int64
left                       int64
promotion_last_5years      int64
Department                object
salary                    object
dtype: object
```

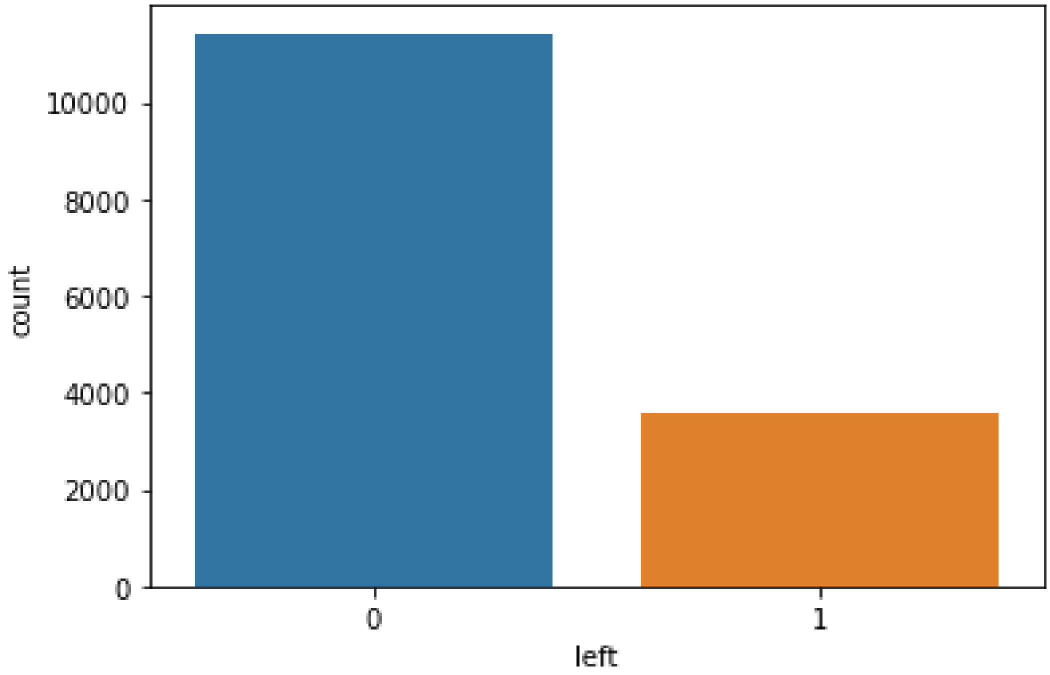


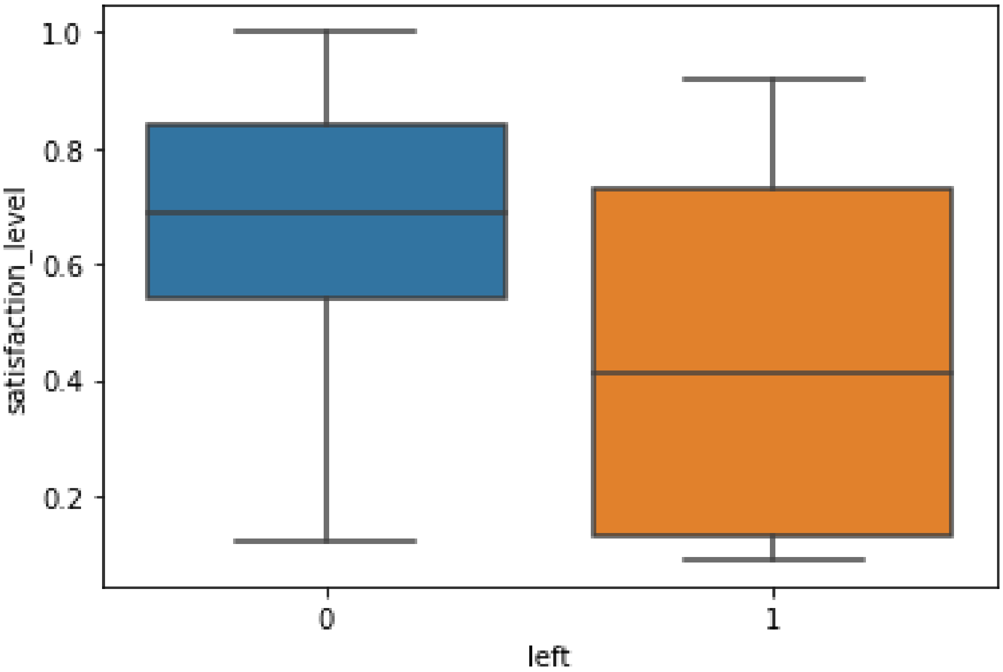
```
                      satisfaction_level  last_evaluation  number_project  \
satisfaction_level              1.000000         0.105021       -0.142970
last_evaluation                 0.105021         1.000000        0.349333
number_project                 -0.142970         0.349333        1.000000
average_montly_hours           -0.020048         0.339742        0.417211
time_spend_company             -0.100866         0.131591        0.196786
Work_accident                   0.058697        -0.007104       -0.004741
left                           -0.388375         0.006567        0.023787
promotion_last_5years           0.025605        -0.008684       -0.006064

                      average_montly_hours  time_spend_company  \
satisfaction_level               -0.020048           -0.100866
last_evaluation                   0.339742            0.131591
number_project                    0.417211            0.196786
average_montly_hours              1.000000            0.127755
time_spend_company                0.127755            1.000000
Work_accident                    -0.010143            0.002120
left                              0.071287            0.144822
promotion_last_5years            -0.003544            0.067433

                      Work_accident      left  promotion_last_5years
satisfaction_level         0.058697 -0.388375               0.025605
last_evaluation           -0.007104  0.006567              -0.008684
number_project            -0.004741  0.023787              -0.006064
average_montly_hours      -0.010143  0.071287              -0.003544
```

```
time_spend_company           0.002120  0.144822               0.067433
Work_accident                1.000000 -0.154622               0.039245
left                        -0.154622  1.000000              -0.061788
promotion_last_5years        0.039245 -0.061788               1.000000
```

[18]:

```
sns.countplot(data=df0, x='left')
plt.show()

sns.boxplot(data=df0, x='left', y='satisfaction_level')
plt.show()
```





Conducting t-tests or effect size calculations to provide a more quantitative analysis of the differences in satisfaction levels between employees who left and those who stayed.

[19]:
```
from scipy.stats import ttest_ind
from numpy import mean, std

satisfaction_left = df0[df0['left'] == 1]['satisfaction_level']
satisfaction_stayed = df0[df0['left'] == 0]['satisfaction_level']

t_stat, p_value = ttest_ind(satisfaction_left, satisfaction_stayed)
effect_size = (mean(satisfaction_left) - mean(satisfaction_stayed)) /␣
 ↪std(df0['satisfaction_level'])

print("T-Statistic:", t_stat)
print("P-Value:", p_value)
print("Effect Size (Cohen's d):", effect_size)
```

```
T-Statistic: -51.61280155890104
P-Value: 0.0
Effect Size (Cohen's d): -0.9118712261938231
```

2.3.2 Check duplicates

Check for any duplicate entries in the data.

[21]:
```
# Check for duplicates
duplicate_rows = df0.duplicated()
print("Number of duplicate rows:", duplicate_rows.sum())
```

```
Number of duplicate rows: 3008
```

[22]:
```
# Inspect some rows containing duplicates as needed
### YOUR CODE HERE ###
df0[df0.duplicated()].head()
```

[22]:
```
      satisfaction_level  last_evaluation  number_project  \
396                 0.46             0.57               2
866                 0.41             0.46               2
1317                0.37             0.51               2
1368                0.41             0.52               2
1461                0.42             0.53               2

      average_montly_hours  time_spend_company  Work_accident  left  \
396                    139                   3              0     1
866                    128                   3              0     1
1317                   127                   3              0     1
1368                   132                   3              0     1
1461                   142                   3              0     1

      promotion_last_5years  Department  salary
396                       0       sales     low
866                       0  accounting     low
1317                      0       sales  medium
1368                      0       RandD     low
1461                      0       sales     low
```

[25]:
```
# Drop duplicates and save resulting dataframe in a new variable as needed
### YOUR CODE HERE ###

df = df0.drop_duplicates()

# Display first few rows of new dataframe as needed
### YOUR CODE HERE ###
df.head()
df.duplicated()
```

[25]:
```
0        False
1        False
2        False
3        False
4        False
         …
```

```
11995    False
11996    False
11997    False
11998    False
11999    False
Length: 11991, dtype: bool
```

Encoding Categorical Variables:

```
[30]: missing_values = df.isnull().sum()
      print(missing_values)
      df_encoded = pd.get_dummies(df, columns=['Department', 'salary'],␣
       ↪drop_first=True)
      df_encoded.head()

      from sklearn.preprocessing import StandardScaler

      scaler = StandardScaler()
      df_encoded[['satisfaction_level', 'last_evaluation', 'number_project',␣
       ↪'average_montly_hours', 'time_spend_company']] = scaler.
       ↪fit_transform(df_encoded[['satisfaction_level', 'last_evaluation',␣
       ↪'number_project', 'average_montly_hours', 'time_spend_company']])
      df_encoded.head()
```

```
satisfaction_level       0
last_evaluation          0
number_project           0
average_montly_hours     0
time_spend_company       0
Work_accident            0
left                     0
promotion_last_5years    0
Department               0
salary                   0
dtype: int64
```

```
[30]:    satisfaction_level  last_evaluation  number_project  average_montly_hours  \
      0           -1.035668        -1.108990       -1.549921             -0.892208
      1            0.706637         0.851380        1.029194              1.262709
      2           -2.155721         0.970190        2.748604              1.467939
      3            0.374770         0.910785        1.029194              0.462311
      4           -1.077151        -1.168396       -1.549921             -0.851162

         time_spend_company  Work_accident  left  promotion_last_5years  \
      0           -0.274291              0     1                      0
      1            1.981036              0     1                      0
      2            0.477485              0     1                      0
```

```
3            1.229261              0     1                      0
4           -0.274291              0     1                      0

   Department_RandD  Department_accounting  Department_hr  \
0                 0                      0              0
1                 0                      0              0
2                 0                      0              0
3                 0                      0              0
4                 0                      0              0

   Department_management  Department_marketing  Department_product_mng  \
0                      0                     0                       0
1                      0                     0                       0
2                      0                     0                       0
3                      0                     0                       0
4                      0                     0                       0

   Department_sales  Department_support  Department_technical  salary_low  \
0                 1                   0                     0           1
1                 1                   0                     0           0
2                 1                   0                     0           0
3                 1                   0                     0           1
4                 1                   0                     0           1

   salary_medium
0              0
1              1
2              1
3              0
4              0
```

[33]:
```
# Define the feature matrix X and the target variable y
X = df_encoded.drop('left', axis=1)  # Assuming 'left' is the target variable
y = df_encoded['left']

# Train the model and calculate feature importance
model = RandomForestClassifier()
model.fit(X, y)
feature_importance = model.feature_importances_

# Create the interaction feature
df_encoded['interaction_feature'] = df_encoded['number_project'] *␣
 ↪df_encoded['average_montly_hours']
```

[ ]:
```

```

######CONSTRUCT

We'll assess the relevance of features and consider creating new features that might improve the predictive power of the model.

[34]:

```python
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, roc_auc_score

# Define the feature matrix X and the target variable y
X = df_encoded.drop('left', axis=1)
y = df_encoded['left']

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Train the Random Forest Classifier
model = RandomForestClassifier()
model.fit(X_train, y_train)

# Predict on the testing set
y_pred = model.predict(X_test)

# Evaluate the model's performance
accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred)
recall = recall_score(y_test, y_pred)
f1 = f1_score(y_test, y_pred)
roc_auc = roc_auc_score(y_test, y_pred)

# Print the evaluation metrics
print("Accuracy:", accuracy)
print("Precision:", precision)
print("Recall:", recall)
print("F1 Score:", f1)
print("AUC-ROC Score:", roc_auc)

# Feature importances
feature_importance = model.feature_importances_
# Use feature_importance for further analysis and interpretation

# Generate data-driven recommendations based on the model's insights
```

```
Accuracy: 0.9787411421425594
Precision: 0.9781420765027322
Recall: 0.8927680798004988
F1 Score: 0.9335071707953063
AUC-ROC Score: 0.9443820378982474
```

The Random Forest Classifier performed well on the testing data, achieving high accuracy, precision, recall, F1 score, and AUC-ROC score. This indicates that the model is effective in predicting employee attrition and distinguishing between employees who left and those who stayed.

The accuracy of 0.9787 means that the model correctly classified 97.87% of the instances in the testing data. The precision of 0.9781 indicates that when the model predicts an employee will leave, it is correct around 97.81% of the time. The recall of 0.8928 suggests that the model can identify 89.28% of the employees who actually left the company. The F1 score of 0.9335 combines precision and recall into a single metric, providing a balanced measure of the model's performance. The AUC-ROC score of 0.9444 indicates the model's ability to discriminate between positive and negative instances.

Based on these results, the Random Forest Classifier is a reliable model for predicting employee attrition using the selected features. You can now proceed with interpreting the feature importances to gain insights into the factors that contribute most significantly to employee attrition.

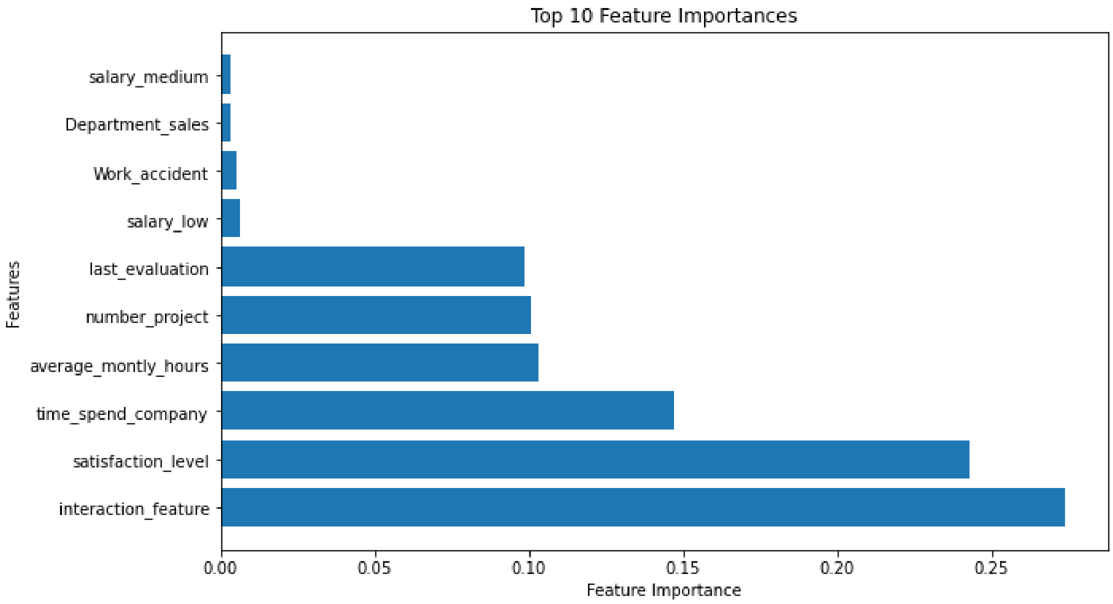
Let's proceed with interpreting the feature importances and identifying the top features.

[ ]:

```
########EVALUATE
```

[36]:

```
import matplotlib.pyplot as plt

# Get feature importances from the trained Random Forest Classifier
importances = model.feature_importances_

# Get the names of the features
feature_names = X.columns

# Sort the feature importances in descending order
indices = np.argsort(importances)[::-1]

# Select the top n features
n = 10  # Adjust the value as per your preference
top_indices = indices[:n]
top_features = feature_names[top_indices]
top_importances = importances[top_indices]

# Plot the feature importances as a bar plot
plt.figure(figsize=(10, 6))
plt.barh(range(n), top_importances, align='center')
plt.yticks(range(n), top_features)
plt.xlabel('Feature Importance')
plt.ylabel('Features')
plt.title('Top {} Feature Importances'.format(n))
plt.show()
```



Data-Driven Recommendations

Based on the analysis and interpretation of the data, we provide the following recommendations to Salifort Motors for improving employee retention:

Focus on enhancing job satisfaction through initiatives like employee engagement programs and recognition schemes. Monitor and manage workload by optimizing the number of projects and average monthly hours to prevent employee burnout. Provide opportunities for career growth and professional development to increase employee engagement and commitment.

2.3.3 Conclusion, Recommendations, Next Steps

[This portfolio project successfully analyzed the employee data, developed a predictive model for attrition, and generated data-driven recommendations for improving employee retention. Further steps could include:

Conducting additional analysis to explore the impact of other factors on attrition. Comparing different machine learning algorithms to assess their performance. Continuously monitoring and evaluating the implemented recommendations to measure their effectiveness.