In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.impute import SimpleImputer
from sklearn.feature_selection import SelectKBest, f_classif
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, c
import warnings
warnings.filterwarnings('ignore')

# 加載數據
df = pd.read_csv("C:/Users/user/Desktop/Cathay/data/loan_data.csv")
print(df.head())
print(df.shape)
```

```
   person_age person_gender person_education  person_income  person_emp_exp  \
0        22.0        female           Master        71948.0               0
1        21.0        female      High School        12282.0               0
2        25.0        female      High School        12438.0               3
3        23.0        female         Bachelor        79753.0               0
4        24.0          male           Master        66135.0               1

  person_home_ownership  loan_amnt loan_intent  loan_int_rate  \
0                  RENT    35000.0    PERSONAL          16.02
1                   OWN     1000.0   EDUCATION          11.14
2              MORTGAGE     5500.0     MEDICAL          12.87
3                  RENT    35000.0     MEDICAL          15.23
4                  RENT    35000.0     MEDICAL          14.27

   loan_percent_income  cb_person_cred_hist_length  credit_score  \
0                 0.49                         3.0           561
1                 0.08                         2.0           504
2                 0.44                         3.0           635
3                 0.44                         2.0           675
4                 0.53                         4.0           586

  previous_loan_defaults_on_file  loan_status
0                             No            1
1                            Yes            0
2                             No            1
3                             No            1
4                             No            1
(45000, 14)
```

In [2]:
```
# 檢查數據基本信息
print("數據基本信息：")
print(df.info())
print("\n基本統計描述：")
print(df.describe())
print("\n缺失值檢查：")
print(df.isnull().sum())

# 檢查目標變量的分布
```

```
print("\n目標變量分布：")
print(df['loan_status'].value_counts(normalize=True))
```

數據基本信息：

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 45000 entries, 0 to 44999
Data columns (total 14 columns):
 #   Column                          Non-Null Count  Dtype  
---  ------                          --------------  -----  
 0   person_age                      45000 non-null  float64
 1   person_gender                   45000 non-null  object 
 2   person_education                45000 non-null  object 
 3   person_income                   45000 non-null  float64
 4   person_emp_exp                  45000 non-null  int64  
 5   person_home_ownership           45000 non-null  object 
 6   loan_amnt                       45000 non-null  float64
 7   loan_intent                     45000 non-null  object 
 8   loan_int_rate                   45000 non-null  float64
 9   loan_percent_income             45000 non-null  float64
 10  cb_person_cred_hist_length      45000 non-null  float64
 11  credit_score                    45000 non-null  int64  
 12  previous_loan_defaults_on_file  45000 non-null  object 
 13  loan_status                     45000 non-null  int64  
dtypes: float64(6), int64(3), object(5)
memory usage: 4.8+ MB
None
```

基本統計描述：

	person_age	person_income	person_emp_exp	loan_amnt
count	45000.000000	4.500000e+04	45000.000000	45000.000000
mean	27.764178	8.031905e+04	5.410333	9583.157556
std	6.045108	8.042250e+04	6.063532	6314.886691
min	20.000000	8.000000e+03	0.000000	500.000000
25%	24.000000	4.720400e+04	1.000000	5000.000000
50%	26.000000	6.704800e+04	4.000000	8000.000000
75%	30.000000	9.578925e+04	8.000000	12237.250000
max	144.000000	7.200766e+06	125.000000	35000.000000

	loan_int_rate	loan_percent_income	cb_person_cred_hist_length
count	45000.000000	45000.000000	45000.000000
mean	11.006606	0.139725	5.867489
std	2.978808	0.087212	3.879702
min	5.420000	0.000000	2.000000
25%	8.590000	0.070000	3.000000
50%	11.010000	0.120000	4.000000
75%	12.990000	0.190000	8.000000
max	20.000000	0.660000	30.000000

	credit_score	loan_status
count	45000.000000	45000.000000
mean	632.608756	0.222222
std	50.435865	0.415744
min	390.000000	0.000000
25%	601.000000	0.000000
50%	640.000000	0.000000
75%	670.000000	0.000000
max	850.000000	1.000000

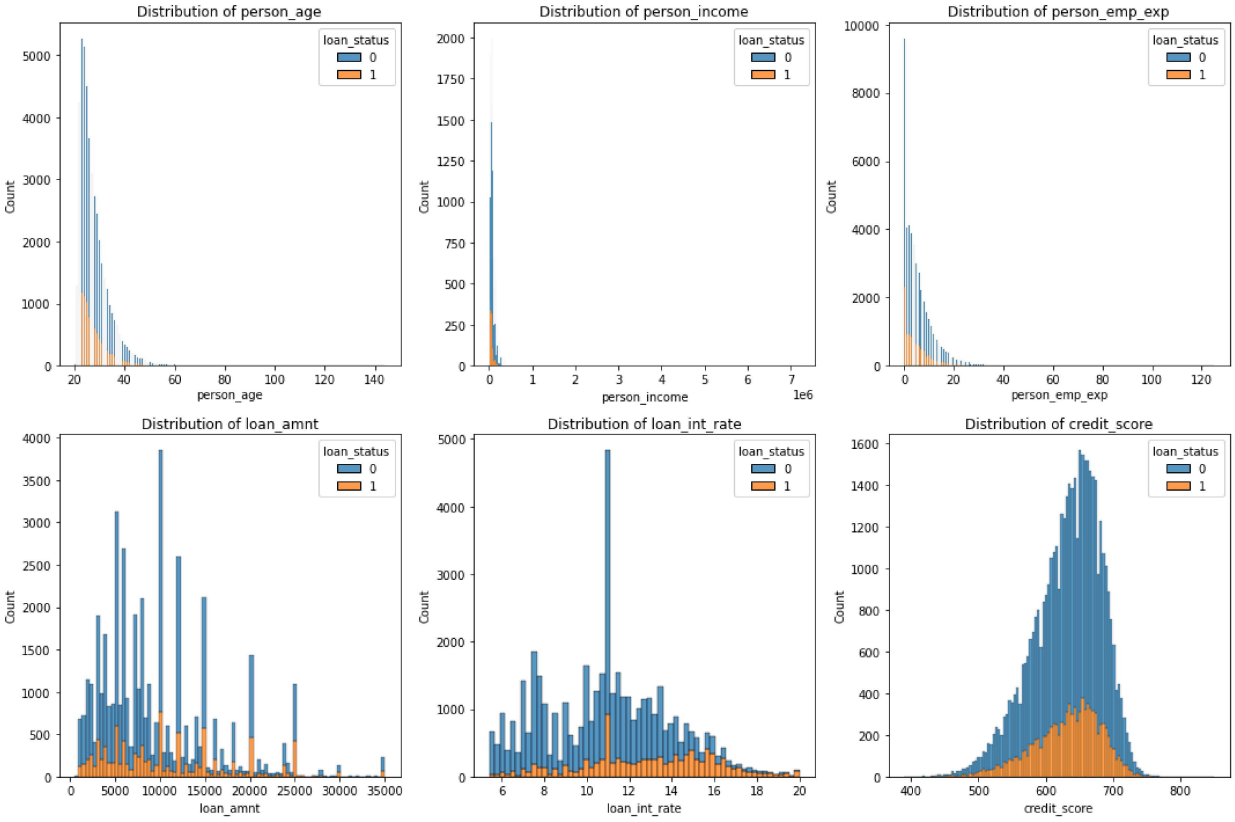
缺失值檢查：

```
person_age                      0
person_gender                   0
person_education                0
person_income                   0
```

```
person_emp_exp                  0
person_home_ownership           0
loan_amnt                       0
loan_intent                     0
loan_int_rate                   0
loan_percent_income             0
cb_person_cred_hist_length      0
credit_score                    0
previous_loan_defaults_on_file  0
loan_status                     0
dtype: int64

目標變量分布：
0    0.777778
1    0.222222
Name: loan_status, dtype: float64
```

In [3]:

```python
# 創建數值型特徵的分布圖
numerical_features = ['person_age', 'person_income', 'person_emp_exp',
                      'loan_amnt', 'loan_int_rate', 'credit_score']

plt.figure(figsize=(15, 10))
for i, feature in enumerate(numerical_features, 1):
    plt.subplot(2, 3, i)
    sns.histplot(data=df, x=feature, hue='loan_status', multiple="stack")
    plt.title(f'Distribution of {feature}')
plt.tight_layout()
plt.show()

# 計算數值特徵與目標變量的相關性
correlation_matrix = df[numerical_features + ['loan_status']].corr()
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', center=0)
plt.title('Correlation Matrix')
plt.show()
```


Distribution of person_age
loan_status
0
1
Count
person_age
Distribution of person_income
person_income
1e6
Distribution of person_emp_exp
person_emp_exp
Distribution of loan_amnt
loan_amnt
Distribution of loan_int_rate
loan_int_rate
Distribution of credit_score
credit_score


Correlation Matrix
person_age
person_income
person_emp_exp
loan_amnt
loan_int_rate
credit_score
loan_status
1 0.19 0.95 0.051 0.013 0.18 -0.021
0.19 1 0.19 0.24 0.0015 0.036 -0.14
0.95 0.19 1 0.045 0.017 0.19 -0.02
0.051 0.24 0.045 1 0.15 0.0091 0.11
0.013 0.0015 0.017 0.15 1 0.011 0.33
0.18 0.036 0.19 0.0091 0.011 1 -0.0076
-0.021 -0.14 -0.02 0.11 0.33 -0.0076 1
1.0
0.8
0.6
0.4
0.2
0.0

In [ ]: