银行客户流失SVM

数据预处理部分

1. 只使用部分相关特征

```
features = ['Age', 'CreditScore', 'EB', 'EstimatedSalary', 'Exited',
       'Gender', 'Geography', 'HasCrCard', 'IsActiveMember',
'NumOfProducts',
       'Tenure']
df = df[features]
df_test = df_test[features]
```

2. 标准化

```
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

SVM过程

- 初始化SVM模型

```
svm_model = SVC(probability=True, random_state=42)
```

- 10-折交叉验证在训练集上建模

```
start_time = time.time()
cv_accuracy = cross_val_score(svm_model, X_train, y_train, cv=10,
scoring='accuracy')
end_time = time.time()
total_time = end_time - start_time
```

  最终用时59.8214 seconds

- 在测试集上测试

```
svm_model.fit(X_train_scaled, y_train)
```

最终得到Accuracy: 0.7690，AUC: 0.7784，F1 Score: 0.5882，Recall: 0.6346，Precision: 0.5482

加速后的SVM

- 首先引入OneAPI的sklearn

```
from sklearnex import patch_sklearn
patch_sklearn()
```

- 然后导入sklearnex中的SVM模型

```
from sklearnex.svm import SVC
```

- 其余过程和普通SVM一样，最终结果如下：
  - 耗时11.8947 seconds，相比普通SVM（59.8214 seconds）速度大幅提升
  - ACC/AUC/F1/Recall/Precision则和普通SVM差不多
  - 可见OneAPI可以显著提升计算速度，但对结果的提升相对有限