

creditcard.csv를 불러온 결과,  
데이터의 총 개수는 284807개,  
매개변수는 Time, V1부터 V28, Amount, Class (사기 여부)가 존재했고,  
사기가 아닌 거래 (Class 0)는 전체의 99.3273%, 사기인 거래 (Class 1)은 0.1727%이다.

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V23	V24	V25	V26	V27	V28	Amount	Class
0	0.0	-1.359807	-0.072781	2.536347	1.378155	-0.338321	0.462388	0.239599	0.098698	0.363787	...	-0.018307	0.277838	-0.110474	0.066928	0.128539	-0.189115	0.133558	-0.021053	149.62	0
1	0.0	1.191857	0.266151	0.166480	0.448154	0.060018	-0.082361	-0.078803	0.085102	-0.255425	...	-0.225775	-0.638672	0.101288	-0.339846	0.167170	0.125895	-0.008983	0.014724	2.69	0
2	1.0	-1.358354	-1.340163	1.773209	0.379780	-0.503198	1.800499	0.791461	0.247676	-1.514654	...	0.247998	0.771679	0.909412	-0.689281	-0.327642	-0.139097	-0.055353	-0.059752	378.66	0
3	1.0	-0.966272	-0.185226	1.792993	-0.863291	-0.010309	1.247203	0.237609	0.377436	-1.387024	...	-0.108300	0.005274	-0.190321	-1.175575	0.647376	-0.221929	0.062723	0.061458	123.50	0
4	2.0	-1.158233	0.877737	1.548718	0.403034	-0.407193	0.095921	0.592941	-0.270533	0.817739	...	-0.009431	0.798278	-0.137458	0.141267	-0.206010	0.502292	0.219422	0.215153	69.99	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
284802	172786.0	-11.881118	10.071785	-9.834783	-2.066656	-5.364473	-2.606837	-4.918215	7.305334	1.914428	...	0.213454	0.111864	1.014480	-0.509348	1.436807	0.250034	0.943651	0.823731	0.77	0
284803	172787.0	-0.732789	-0.055080	2.035030	-0.738589	0.868229	1.058415	0.024330	0.294869	0.584800	...	0.214205	0.924384	0.012463	-1.016226	-0.606624	-0.395255	0.068472	-0.053527	24.79	0
284804	172788.0	1.919565	-0.301254	-3.249640	-0.557828	2.630515	3.031260	-0.296827	0.708417	0.432454	...	0.232045	0.578229	-0.037501	0.640134	0.265745	-0.087371	0.004455	-0.026561	67.88	0
284805	172788.0	-0.240440	0.530483	0.702510	0.689799	-0.377961	0.623708	-0.686180	0.679145	0.392087	...	0.265245	0.800049	-0.163298	0.123205	-0.569159	0.546668	0.108821	0.104533	10.00	0
284806	172792.0	-0.533413	-0.189733	0.703337	-0.506271	-0.012546	-0.649617	1.577006	-0.414650	0.486180	...	0.261057	0.643078	0.376777	0.008797	-0.473649	-0.818267	-0.002415	0.013649	217.00	0

사기가 아닌 거래 데이터의 개수가 너무 많아, 전체 데이터 중 10000개만 추출하여 새로운 데이터를 만듭니다. 이제 사기가 아닌 거래는 전체의 95.3107%, 사기인 거래는 4.6893%입니다. 이제 Amount 변수의 표준화를 진행합니다. 원본의 dataframe에서 Amount를 제거한 뒤, Amount\_Scaled라는 새로운 열을 추가하여, 해당 열은 기존 Amount 열이 표준화된 값으로 재지정합니다. 이후 X, y로 데이터를 분리하였습니다.

이제 전체의 20%를 test 데이터, 80%를 train 데이터로 분리합니다. train 데이터는 8393개, test 데이터는 2099개가 존재합니다.

train 데이터에서 Class 1의 개수가 매우 적어 과적합의 위험이 존재합니다. 따라서 SMOTE를 통해 Class 1 데이터의 개수를 Class 0 개수와 동일하도록 만들어줍니다. 전체데이터의 개수는 15998개입니다.

이제 skleran의 Logistic Regression 모델을 활용하여 데이터를 학습합니다. max\_iter는 10000으로 설정하였습니다. 결과는 다음과 같습니다.

```
y_pred: [0 0 0 0 0 0 1 0 0 0]
y_proba: [0.13831676 0.07582168 0.04098245 0.02229752 0.26721734 0.0019319
1. 0.00679588 0.00119367 0.0137785 ]
          precision    recall  f1-score   support

         0           1.00         0.99         0.99         2001
         1           0.81         0.93         0.87           98

 accuracy          0.99         0.99         0.99         2099
 macro avg          0.90         0.96         0.93         2099
weighted avg          0.99         0.99         0.99         2099

PR-AUC: 0.9552035824854457
```

주어진 제한 조건을 모두 만족하는 것을 확인할 수 있었습니다.