

Iris Dataset - Petal Length ANOVA 분석 Report

1. 데이터 불러오기 및 구조 확인

```
python import seaborn as sns iris = sns.load_dataset('iris') iris.head() iris.info()
```

iris는 150개의 관측치와 5개의 변수(sepal_length, sepal_width, petal_length, petal_width, species)로 구성되어있다. species는 setosa, versicolor, virginica의 3가지 품종을 나타낸다.

2. 기술통계량 산출

```
iris.groupby('species')['petal_length'].describe() iris['species'].value_counts()
```

각 그룹의 평균, 표준편차, 최소/최대값, 사분위수를 확인한다. 그룹별 데이터 수는 각 50개씩으로, 균형 잡힌 데이터셋임을 알 수 있었다.

3. Boxplot 시각화

```
import seaborn as sns import matplotlib.pyplot as plt
```

```
sns.boxplot(x='species', y='petal_length', data=iris) plt.title("Petal Length by Species") plt.show()
```

Setosa는 가장 짧고, Virginica는 가장 긴 Petal Length를 보였고, Versicolor는 중간값을 가짐을 확인하였다. 그룹 간 차이가 명확하게 보임을 알 수 있다.

4. 정규성 검정 (Shapiro-Wilk)

```
''' from scipy.stats import shapiro
```

```
for species in iris['species'].unique(): stat, p = shapiro(iris[iris['species'] == species]
['petal_length']) print(f"{species}: p-value = {p:.4f}") ''' 가설: H0: 각 그룹의 Petal Length
는 정규분포를 따른다 H1: 정규분포를 따르지 않는다
```

해석: 모든 그룹에서 p-value > 0.05 이므로, 정규성 가정을 만족한다.

5. 등분산성 검정 (Levene Test)

```
from scipy.stats import levene
```

```
group1 = iris[iris['species'] == 'setosa']['petal_length'] group2 =  
iris[iris['species'] == 'versicolor']['petal_length'] group3 = iris[iris['species'] ==  
'virginica']['petal_length']
```

```
stat, p = levene(group1, group2, group3) print(f"Levene Test p-value = {p:.4f}")
```

가설: H_0 : 세 그룹은 등분산이다

H_1 : 적어도 한 그룹은 분산이 다르다

해석: p-value > 0.05이므로, 등분산 가정을 만족한다.

6. 가설 수립

H_0 : 3개 Species의 Petal Length 평균은 모두 같다

H_1 : 적어도 한 Species의 평균은 다르다

7. One-Way ANOVA 분석

```
from scipy.stats import f_oneway
```

```
f_stat, p = f_oneway(group1, group2, group3) print(f"F = {f_stat:.4f}, p-value =  
{p:.4e}")
```

해석: F값이 크고 p-value < 0.05 이므로, 귀무가설을 기각한다.

⇒ 세 그룹 간 유의미한 평균 차이가 존재한다.

8. 사후검정 (Tukey HSD)

```
from statsmodels.stats.multicomp import pairwise_tukeyhsd
```

```
tukey = pairwise_tukeyhsd(endog=iris['petal_length'], groups=iris['species'],  
alpha=0.05) print(tukey)
```

해석: 모든 쌍(Setosa vs Versicolor, Setosa vs Virginica, Versicolor vs Virginica) 간 유의미한 차이가 존재한다. ($p < 0.05$)

9. 최종 결론

1. Virginica > Versicolor > Setosa 순으로 Petal Length 평균이 크다.
2. ANOVA 및 사후검정 결과, 모든 품종 간 Petal Length에서 통계적으로 유의한 차이가 존재한다.

3. Boxplot과 일관된 결과를 보이며, 정규성/등분산성 가정을 충족한 믿을 수 있는 분석이다.

Credit Card Fraud Detection Report

1. 프로젝트 개요

- 목표: 신용카드 거래 데이터에서 사기 거래(Class=1)를 정확히 탐지하는 모델을 개발한다.
- 데이터셋: Kaggle의 `creditcard.csv`
- 특징: 극심한 클래스 불균형 (Class 0: 정상 거래, Class 1: 사기 거래)

2. 분석 과정 요약

2.1 데이터 전처리

- `Amount` 변수만 `StandardScaler` 로 표준화 → `Amount_Scaled` 생성
- 사기 거래(Class=1)는 전부 유지, 정상 거래(Class=0)는 10,000건 샘플링하여 비율 완화
- `X`, `y` 로 분리

2.2 데이터 분할 및 SMOTE 적용

- `train_test_split` (학습:테스트 = 8:2, stratify 적용)
- 학습 데이터에 **SMOTE** 적용해 Class 1을 오버샘플링
- 🚩 **SMOTE 이유**: 모델이 소수 클래스(Class=1)를 충분히 학습하도록 지원

2.3 모델 학습 및 평가




- 모델: Logistic Regression
- 평가지표: Precision, Recall, F1-score, PR-AUC
- 초기 결과: Recall과 F1 점수가 목표치에 미달

3. Threshold 조정 결과

Threshold	Recall (Class 1)	F1-score (Class 1)	PR-AUC
0.80	0.8673	0.8995	0.9163
0.75	0.8673	0.8947	0.9163
0.60~0.70	0.8673	0.8673~0.8763	0.9163

- 🔍 Threshold=0.80에서 목표 달성

4. 최종 성능 평가

- **Recall ≥ 0.80 :**  0.8673
 - **F1-score ≥ 0.88 :**  0.8995
 - **PR-AUC ≥ 0.90 :**  0.9163
-

5. 결론 및 제안

- SMOTE와 Threshold 조정은 불균형 문제 해결에 효과적이다.
 - 단순 모델(Logistic Regression)로도 목표 성능에 달성하였다.
 - 추가 제안:
 - XGBoost, RandomForest 등 더 복잡한 모델을 활용해볼 수 있다.
 - 클래스별 비용 고려한 Cost-sensitive Learning 를 시도해볼 수 있다.
-