분석 예제 데이터

Fisher 분꽃 데이터

> Target 집단 : 3개 종 Iris setosa, Iris virginica and Iris versicolor 판별변수 : Sepal (꽃받침) 넓이, 길이 / Petal 꽃잎 넓이, 길이

```
import pandas as pd
df=pd.read_csv('http://wolfpack.hnu.ac.kr/Stat_Notes/example_data/iris.csv')
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
Sepal_Length    150 non-null int64
Sepal_Width     150 non-null int64
Petal_Length    150 non-null int64
Petal_Width     150 non-null int64
group           150 non-null object
dtypes: int64(4), object(1)
memory usage: 5.9+ KB
```

상금 순위변수 만들고 집단변수 만들기

BOX M-test []

> 집단 간 공분산 동등성 검증

> 공분산 동일하면 선형 판별분석, 이분산이면 이차판별분석 실시

> 파이썬에는 이를 제공하지 않음

```
import rpy2
%load_ext rpy2.ipython
```

```
%%R
iris<-read.csv('http://wolfpack.hnu.ac.kr/Stat_Notes/example_data/iris.csv')
str(iris)
```

```
'data.frame':   150 obs. of  5 variables:
 $ Sepal_Length: int  50 64 65 67 63 46 69 62 59 46 ...
 $ Sepal_Width : int  33 28 28 31 28 34 31 22 32 36 ...
 $ Petal_Length: int  14 56 46 56 51 14 51 45 48 10 ...
 $ Petal_Width : int  2 22 15 24 15 3 23 15 18 2 ...
 $ group       : Factor w/ 3 levels "Setosa","Versicolor",..: 1 3 2 3 3 1 3 2 2 1 ...
```

```
%%R
#install.packages('heplots')
library(heplots)
boxM(iris[,1:4],iris[,5])
```

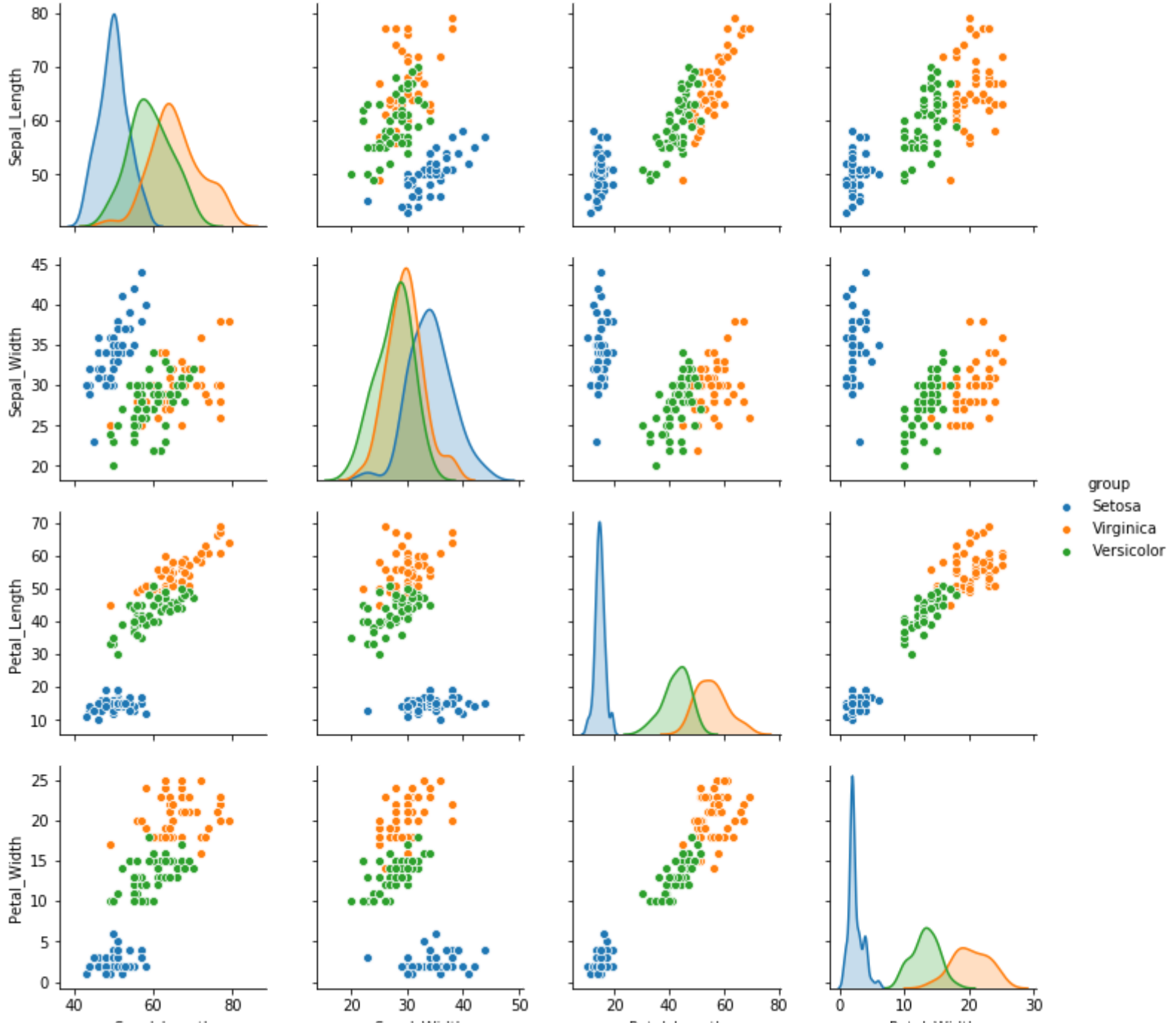
```
        Box's M-test for Homogeneity of Covariance Matrices

data:  iris[, 1:4]
Chi-Sq (approx.) = 140.94, df = 20, p-value < 2.2e-16
```

> 등분산 가정이 만족하지 않아 이차 판별분석을 적용

Scatter plot(판별변수) by Group

```
import seaborn as sns
sns.pairplot(df, hue='group')
```

```
<seaborn.axisgrid.PairGrid at 0x7f296c97d630>
```



Sepal_Length Sepal_Width Petal_Length Petal_Width

Fisher Qurdratic Discriminant analysis [피셔 2차 판별분석]

> QDA() 함수에 의해 선형판별분석 결과가 저장된다

```
import numpy as np
from sklearn.discriminant_analysis import QuadraticDiscriminantAnalysis as QDA
X=df.iloc[:,0:4].values
y=df.iloc[:,4].values
qda=QDA()
X_lda=qda.fit(X, y)
```

집단 소속확률

```
qda.predict_proba(X)[0:3] #Estimate probability
```

```
array([[1.00000000e+000, 5.40881752e-023, 2.33885374e-038],
       [1.48454192e-165, 1.83495222e-007, 9.99999817e-001],
       [1.75567847e-090, 9.97343757e-001, 2.65624318e-003]])
```

> > 첫번째 분꽃은 Setosa 속할 확률 100%임 판별규칙에 의한 판별결과는 qda.predict(X)에 저장되어 있음

추정판별집단 데이터프레임 : 판별예측결과는 np.array 형식으로 저장되어 있어 데이터프레임으로 변환

```
y_pred=pd.DataFrame(qda.predict(X))   #Predict class labels for samples in X.
y_pred.columns=['group_qda']
y_pred.head(3)
```

	group_qda
0	Setosa
1	Virginica
2	Versicolor

[원데이터+판별집단] 합치기 - 정분류 교차표

```
df_qda=pd.concat([df,y_pred],axis=1)
df_qda['DA_result']=df_qda.group+'-'+df_qda.group_qda
qda_table=pd.crosstab(df_qda.group,df_qda.group_qda)
qda_table
```

group_qda	Setosa	Versicolor	Virginica
group			
Setosa	50	0	0
Versicolor	0	48	2
Virginica	0	1	49

```
qda_table.apply(lambda r: r/r.sum(), axis=1)  #정분류
```

group_qda	Setosa	Versicolor	Virginica
group			
Setosa	1.0	0.00	0.00
Versicolor	0.0	0.96	0.04
Virginica	0.0	0.02	0.98

> 피셔 선형 판별규칙 정분류 - dn집단 정분류 93.2%, up집단 정분류 100%로 완벽함

새로운 분꽃 품종 판별하기

> 꽃받침 길이 45, 넓이 30, 꽃잎 길이=30, 넓이=15인 새로운 분꽃

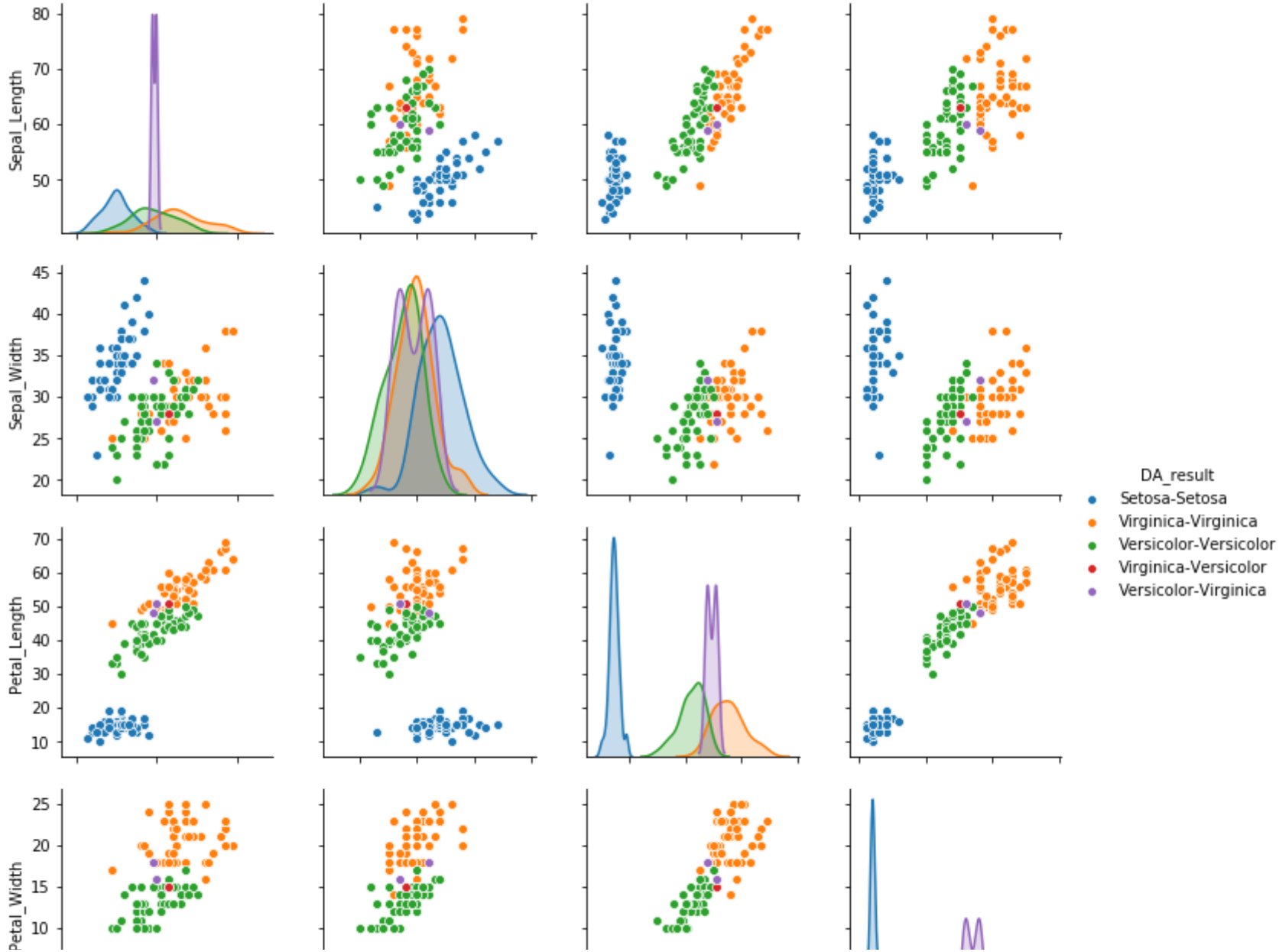
```
new=pd.DataFrame([45,40,30,15]).T
new_X=new.values
print('추정집단확률',qda.predict_proba(new_X),'추정판별집단',qda.predict(new_X))
```

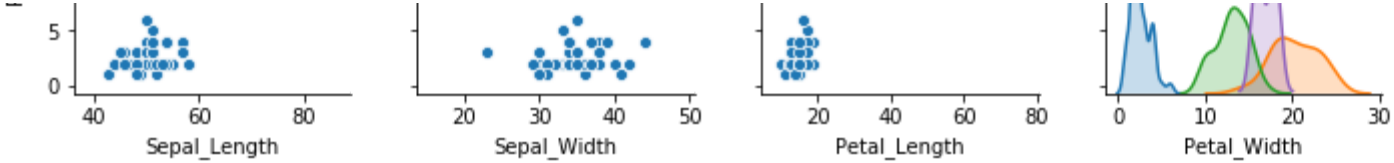
```
추정집단확률 [[6.72252270e-35 9.64390758e-01 3.56092417e-02]] 추정판별집단 ['Versicolor']
```

이차 판별분석 결과 시각화

```
import seaborn as sns
sns.pairplot(df_qda, hue='DA_result')
```

```
/usr/local/lib/python3.6/dist-packages/numpy/core/_methods.py:140: RuntimeWarning: Degrees of freedom <= 0 for slice
  keepdims=keepdims)
/usr/local/lib/python3.6/dist-packages/numpy/core/_methods.py:132: RuntimeWarning: invalid value encountered in doub
  ret = ret.dtype.type(ret / rcount)
<seaborn.axisgrid.PairGrid at 0x7f296526b588>
```





판별변수 (평균, 표준편차) by 판별결과

```
df_qda.groupby(['DA_result']).mean()
```

DA_result	Sepal_Length	Sepal_Width	Petal_Length	Petal_Width
Setosa-Setosa	50.060000	34.28000	14.620000	2.460000
Versicolor-Versicolor	59.354167	27.62500	42.312500	13.104167
Versicolor-Virginica	59.500000	29.50000	49.500000	17.000000
Virginica-Versicolor	63.000000	28.00000	51.000000	15.000000
Virginica-Virginica	65.938776	29.77551	55.612245	20.367347

주성분 활용 판별결과 보기

```
# Standardizing the features
from sklearn.preprocessing import StandardScaler
df_s=StandardScaler().fit_transform(df_qda.iloc[:,0:4])
# PCA
from sklearn.decomposition import PCA
pca=PCA(0.8) #80% rule pca=PCA(0.8)
df_pca=pca.fit_transform(df_s) #PC variables
```

```
df_loading=pd.DataFrame(pca.components_.T)  #loading values
df_loading
```

	0	1
0	0.521066	0.377418
1	-0.269347	0.923296
2	0.580413	0.024492
3	0.564857	0.066942

```
df_pca=pd.DataFrame(df_pca,columns=['FL_size','Petal_size'])
df_pca.set_index(df_qda.index,inplace=True)  #주성분점수 데이터프레임 행 인덱스 원 데이터 사용
pca_df=pd.concat([df_qda,df_pca],axis=1)
pca_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 9 columns):
Sepal_Length    150 non-null int64
Sepal_Width     150 non-null int64
Petal_Length    150 non-null int64
Petal_Width     150 non-null int64
group           150 non-null object
group_qda       150 non-null object
DA_result       150 non-null object
FL_size         150 non-null float64
Petal_size      150 non-null float64
dtypes: float64(2), int64(4), object(3)
memory usage: 10.6+ KB
```

```
pca_df0=pca_df.iloc[:,[6,7,8]]
import seaborn as sns
sns.pairplot(pca_df0, hue='DA_result')
```

```
/usr/local/lib/python3.6/dist-packages/numpy/core/_methods.py:140: RuntimeWarning: Degrees of freedom <= 0 for slice
  keepdims=keepdims)
/usr/local/lib/python3.6/dist-packages/numpy/core/_methods.py:132: RuntimeWarning: invalid value encountered in doub
  ret = ret.dtype.type(ret / rcount)
<seaborn.axisgrid.PairGrid at 0x7f2963aef198>
```