14th Mar Assignment

March 17, 2023

1 Assignment 38

Q1. Write a Python function that takes in two arrays of data and calculates the F-value for a variance ratio test. The function should return the F-value and the corresponding p-value for the test.

Ans.

[6]:

```
import numpy as np
from scipy.stats import f

def variance_ratio_test(data1, data2):
    n1 = len(data1)
    n2 = len(data2)
    var1 = np.var(data1, ddof=1)
    var2 = np.var(data2, ddof=1)

    F = var1/var2
    df1 = n1-1
    df2 = n2-1
    p_value = 1-f.cdf(F, df1, df2)

    return F, p_value

data1 = [2,4,6,8,10,12,14,16,18,20]
data2 = [1,3,5,7,9,11,13,15,17,19]
F, p_value = variance_ratio_test(data1, data2)
print(F, p_value)
```

```
1.0 0.5000000000000001
```

Q2. Given a significance level of 0.05 and the degrees of freedom for the numerator and denominator of an F-distribution, write a Python function that returns the critical F-value for a two-tailed test.

Ans.

[7]:

```
from scipy.stats import f
```

```python
def critical_f_value(dfn, dfd, alpha=0.05):
    f_crit = f.ppf(alpha/2, dfn, dfd)

    return f_crit

dfn = 5
dfd = 10
alpha = 0.05
f_crit = critical_f_value(dfn, dfd, alpha)
print(f_crit)
```

```
0.15107670102998205
```

Q3. Write a Python program that generates random samples from two normal distributions with known variances and uses an F-test to determine if the variances are equal. The program should output the F-value, degrees of freedom, and p-value for the test.

Ans.

[8]:

```python
import numpy as np
from scipy.stats import f

# Set the random seed for reproducibility
np.random.seed(1234)

# Generate random samples from two normal distributions with known variances
n1 = 50
n2 = 60
var1 = 4
var2 = 6
mean1 = 10
mean2 = 10.5
sample1 = np.random.normal(mean1, np.sqrt(var1), n1)
sample2 = np.random.normal(mean2, np.sqrt(var2), n2)

# Perform the F-test to determine if the variances are equal
F = np.var(sample1, ddof=1) / np.var(sample2, ddof=1)
dfn = n1 - 1
dfd = n2 - 1
p_value = 2 * (1 - f.cdf(F, dfn, dfd))

# Output the results
print("F-value: ", F)
print("Degrees of freedom: ", dfn, ", ", dfd)
print("p-value: ", p_value)
```

```
F-value:  0.6274376043018689
Degrees of freedom:  49 ,  59
p-value:  1.9049797718924713
```

Q4.The variances of two populations are known to be 10 and 15. A sample of 12 observations is taken from each population. Conduct an F-test at the 5% significance level to determine if the variances are significantly different.

Ans.

[9]:

```python
import numpy as np
from scipy.stats import f

# Set the random seed for reproducibility
np.random.seed(1234)

# Generate random samples from two normal distributions with known variances
var1 = 10
var2 = 15
n1 = 12
n2 = 12
mean1 = 0
mean2 = 0
sample1 = np.random.normal(mean1, np.sqrt(var1), n1)
sample2 = np.random.normal(mean2, np.sqrt(var2), n2)

# Perform the F-test to determine if the variances are equal
F = np.var(sample1, ddof=1) / np.var(sample2, ddof=1)
dfn = n1 - 1
dfd = n2 - 1
alpha = 0.05
p_value = 2 * (1 - f.cdf(F, dfn, dfd))

# Check the p-value against the significance level and output the results
if p_value < alpha:
    print("Reject the null hypothesis, variances are significantly different.")
else:
    print("Fail to reject the null hypothesis, variances are not significantly different.")
print("F-value: ", F)
print("Degrees of freedom: ", dfn, ", ", dfd)
print("p-value: ", p_value)
```

```
Fail to reject the null hypothesis, variances are not significantly different.
F-value:  0.8832878293733425
Degrees of freedom:  11 ,  11
p-value:  1.1593969287689205
```

Q5. A manufacturer claims that the variance of the diameter of a certain product is 0.005. A sample of 25 products is taken, and the sample variance is found to be 0.006. Conduct an F-test at the 1% significance level to determine if the claim is justified.

[10]:

```python
#### Ans.import numpy as np
from scipy.stats import f

# Set the random seed for reproducibility
np.random.seed(1234)

# Generate random samples from a normal distribution with known variance
var = 0.005
n = 25
mean = 0
sample = np.random.normal(mean, np.sqrt(var), n)

# Calculate the sample variance
sample_var = np.var(sample, ddof=1)

# Perform the F-test to determine if the claim is justified
F = sample_var / var
dfn = n - 1
dfd = np.inf
alpha = 0.01
p_value = 1 - f.cdf(F, dfn, dfd)

# Check the p-value against the significance level and output the results
if p_value < alpha:
    print("Reject the null hypothesis, the claim is not justified.")
else:
    print("Fail to reject the null hypothesis, the claim is justified.")
print("F-value: ", F)
print("Degrees of freedom: ", dfn, ", ", dfd)
print("p-value: ", p_value)
```

```
Fail to reject the null hypothesis, the claim is justified.
F-value:  1.0660103731672703
Degrees of freedom:  24 ,  inf
p-value:  1.0
```

Q6. Write a Python function that takes in the degrees of freedom for the numerator and denominator of an F-distribution and calculates the mean and variance of the distribution. The function should return the mean and variance as a tuple.

Ans.

[11]:

```python
def f_dist_mean_var(dfn, dfd):
    if dfd > 2:
        mean = dfd / (dfd - 2)
        var = (2 * dfd ** 2 * (dfn + dfd - 2)) / (dfn * (dfd - 2) ** 2 * (dfd -␣
↪4))
    else:
        mean = float('inf')
        var = float('inf')

    return (mean, var)
```

The mean of an F-distribution is calculated as dfd / (dfd - 2), and the variance is calculated as (2 * dfd^2 * (dfn + dfd - 2)) / (dfn * (dfd - 2)^2 * (dfd - 4)). The function returns the mean and variance as a tuple. You can call this function with the desired values for dfn and dfd and it will return the mean and variance of the F-distribution.

Q7. A random sample of 10 measurements is taken from a normal population with unknown variance. The sample variance is found to be 25. Another random sample of 15 measurements is taken from another normal population with unknown variance, and the sample variance is found to be 20. Conduct an F-test at the 10% significance level to determine if the variances are significantly different.

Ans.

[12]:

```
import numpy as np
from scipy.stats import f

# Sample sizes
n1 = 10
n2 = 15

# Sample variances
s1_sq = 25
s2_sq = 20

# Degrees of freedom
df1 = n1 - 1
df2 = n2 - 1

# F-test
F = s1_sq / s2_sq
p_value = 1 - f.cdf(F, df1, df2)
alpha = 0.1

# Conclusion
if p_value < alpha/2 or p_value > 1 - alpha/2:
    print("Reject null hypothesis: Variances are significantly different")
else:
    print("Fail to reject null hypothesis: Variances are not significantly␣
 ↪different")
```

```
Fail to reject null hypothesis: Variances are not significantly different
```

Q8. The following data represent the waiting times in minutes at two different restaurants on a Saturday night: Restaurant A: 24, 25, 28, 23, 22, 20, 27; Restaurant B: 31, 33, 35, 30, 32, 36. Conduct an F-test at the 5% significance level to determine if the variances are significantly different.

Ans.

[13]:

```
import numpy as np
from scipy.stats import f

# Waiting times for restaurant A and B
a = np.array([24, 25, 28, 23, 22, 20, 27])
b = np.array([31, 33, 35, 30, 32, 36])

# Sample variances
var_a = np.var(a, ddof=1)
var_b = np.var(b, ddof=1)

# Degrees of freedom
df_a = len(a) - 1
df_b = len(b) - 1

# F-test
F = var_a / var_b
p_value = 1 - f.cdf(F, df_a, df_b)
alpha = 0.05

# Conclusion
if p_value < alpha/2 or p_value > 1 - alpha/2:
    print("Reject null hypothesis: Variances are significantly different")
else:
    print("Fail to reject null hypothesis: Variances are not significantly␣
 ↪different")
```

```
Fail to reject null hypothesis: Variances are not significantly different
```

Q9. The following data represent the test scores of two groups of students: Group A: 80, 85, 90, 92, 87, 83; Group B: 75, 78, 82, 79, 81, 84. Conduct an F-test at the 1% significance level to determine if the variances are significantly different.

Ans.

[14]:

```
from scipy.stats import f_oneway

groupA = [80, 85, 90, 92, 87, 83]
groupB = [75, 78, 82, 79, 81, 84]

F, p = f_oneway(groupA, groupB)

print("F-value:", F)
print("p-value:", p)
```

```
F-value: 8.040089086859687
p-value: 0.017684171924487787
```