

Formulario 1: DCA - Parte I

Diseño de experimentos

Andrés Felipe Pico Zúñiga

```
import pandas as pd
import numpy as np
import scipy

from statsmodels.stats.multicomp import pairwise_tukeyhsd

data = "pureza"
```

Enunciado

Problema 1: Una compañía compra materia prima a tres proveedores. La pureza de esta materia prima es un importante en la calidad del producto terminado. El departamento de calidad quiere determinar si existen diferencias en la pureza de la materia prima debido al proveedor. Se selecciona al azar 4 muestras de materia prima de cada proveedor de la última compra realizada en el mes, los resultados fueron los siguientes.

```
df = pd.read_csv(f"./data/{data}.csv")
print(df)
```

```
##      A   B   C
## 0   94  88  95
## 1   95  92  96
## 2   93  86  96
## 3   96  87  92
```

El diseño apropiado para este enunciado es el unifactorial completamente aleatorizado (DCA).

La hipótesis nula es que la pureza de la materia prima no depende del tipo de proveedor.

Solución

Variables iniciales

```
alfa = 0.05           # Nivel de significancia
k = len(df.columns)   # Número de tratamientos (proveedores)
n = len(df)           # Número de muestras por tratamiento
N = k * n             # Número total de observaciones
```

Suma de cuadrados

```
means = df.mean()
total_mean = np.mean(df.values)
```

```
sstr = n * sum((means - total_mean)**2)
print(f"Suma de cuadrados: {sstr}")
```

```
## Suma de cuadrados: 108.5
```

Varianza

```
grados_libertad_tr = k - 1
error = N - k
cmtr = sstr / grados_libertad_tr
print(f"Varianza de los tratamientos: {cmtr}")
```

```
## Varianza de los tratamientos: 54.25
```

F calculada

```
sse = sum(sum((df[col] - means[col])**2) for col in df.columns)
cme = sse / error
f_calculated = cmtr / cme
print(f"F calculada: {f_calculated}")
```

```
## F calculada: 13.376712328767123
```

Valor teórico de F

```
f_teoric = scipy.stats.f.isf(q=alfa, dfn=(k - 1), dfd=(N - k))
print(f"Valor teórico de F: {f_teoric}")
```

```
## Valor teórico de F: 4.25649472909375
```

Conclusión

El valor estadístico F calculado (13.3767123) es considerablemente mayor que el valor F teórico (4.2564947), lo que demuestra que la variación entre los proveedores es significativamente más grande que la variación aleatoria.

Prueba HSD de Tukey

```
alfa = 0.001
df_long = df.melt(var_name='Proveedor', value_name='Pureza')
tukey_results = pairwise_tukeyhsd(
    endog=df_long['Pureza'],
    groups=df_long['Proveedor'],
    alpha=alfa
)

print(tukey_results)
```

```
## Multiple Comparison of Means - Tukey HSD, FWER=0.00
## =====
## group1 group2 meandiff p-adj lower upper reject
## -----
##      A      B    -6.25 0.0045 -14.0718  1.5718  False
```

```
##      A      C      0.25 0.9832 -7.5718  8.0718  False
##      B      C      6.5 0.0035 -1.3218 14.3218  False
## -----
```

```
print("Pureza promedio por proveedor:")
```

```
## Pureza promedio por proveedor:
```

```
print(df.mean().sort_values(ascending=False))
```

```
## C      94.75
## A      94.50
## B      88.25
## dtype: float64
```

De este modo, se ve que los proveedores con mayores valores medios de pureza son A y C. Sin embargo, no hay diferencia significativa entre ellos.

Prueba LSD

```
t_critico = scipy.stats.t.ppf(1 - alfa / 2, df=error)
mse = sse / error
lsd = t_critico * np.sqrt(mse * (2 / n))
```

```
print(f"Cuadrado Medio del Error (MSE): {mse:.4f}")
```

```
## Cuadrado Medio del Error (MSE): 4.0556
```

```
print(f"Valor t-crítico: {t_critico:.4f}")
```

```
## Valor t-crítico: 4.7809
```

```
print(f"Valor LSD con alfa={alfa}: {lsd:.4f}")
```

```
## Valor LSD con alfa=0.001: 6.8080
```

```
print("-" * 50)
```

```
## -----
```

```
print("Comparaciones por pares:")
```

```
## Comparaciones por pares:
```

```
group_means = df.mean().to_dict()
groups = list(group_means.keys())
```

```
for i in range(len(groups)):
    for j in range(i + 1, len(groups)):
        group1 = groups[i]
        group2 = groups[j]
```

```
        mean_diff = abs(group_means[group1] - group_means[group2])
```

```
        print(f"\n- {group1} vs. {group2}:")
        print(f"    Diferencia de medias: {mean_diff:.4f}")
```

```
        if mean_diff > lsd:
            print(f"    Resultado: La diferencia es SIGNIFICATIVA (diferencia > LSD)")
```

```

else:
    print(f" Resultado: La diferencia NO es significativa (diferencia <= LSD)")

##
## - A vs. B:
##   Diferencia de medias: 6.2500
##   Resultado: La diferencia NO es significativa (diferencia <= LSD)
##
## - A vs. C:
##   Diferencia de medias: 0.2500
##   Resultado: La diferencia NO es significativa (diferencia <= LSD)
##
## - B vs. C:
##   Diferencia de medias: 6.5000
##   Resultado: La diferencia NO es significativa (diferencia <= LSD)

```