

# Formulario 2: DCA - Parte II

## 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 = "nitrogeno"
```

## Enunciado

Problema 1: Se realizó una investigación para determinar la pérdida de nitrógeno por transpiración con varios niveles dietéticos de proteínas. En el experimento se utilizaron 16 hombres preadolescentes a quienes se les juzgó estar saludables. Cada muchacho estuvo sujeto a una de las cuatro dietas controladas en las cuales consumía 20 (A), 45 (B), 60 (C) u 85 (D) gramos de proteínas por día. Los siguientes datos representan la pérdida de nitrógeno del cuerpo a través de la transpiración, en miligramos, recabados durante los dos días últimos del periodo de experimentación.

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

```
##      A      B      C      D
## 0  190  210  318  390
## 1  180  240  295  321
## 2  200  250  270  396
## 3  210  235  380  399
```

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

La hipótesis nula es que la pérdida de nitrógeno no depende del tipo de dieta.

## Solución

### Variables iniciales

```
alfa = 0.05          # Nivel de significancia
k = len(df.columns)  # Número de tratamientos
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: 79816.5
```

## 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: 26605.5
```

## 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: 26.224157049570824
```

## 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: 3.490294819497605
```

## Conclusión

El valor estadístico F calculado (26.224157) es considerablemente mayor que el valor F teórico (3.4902948), lo que demuestra que la variación entre las dietas es significativamente más grande que la variación aleatoria.

## Prueba HSD de Tukey

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

print(tukey_results)
```

```
## Multiple Comparison of Means - Tukey HSD, FWER=0.00
## =====
## group1 group2 meandiff p-adj lower upper reject
## -----
```

```
##      A      B      38.75 0.3556 -79.6639 157.1639 False
##      A      C     120.75 0.0008   2.3361 239.1639  True
##      A      D     181.5   0.0   63.0861 299.9139  True
##      B      C      82.0 0.0155 -36.4139 200.4139 False
##      B      D     142.75 0.0002  24.3361 261.1639  True
##      C      D      60.75 0.0795 -57.6639 179.1639 False
## -----
```

```
print("Perdida nitrógeno promedio por dieta:")
```

```
## Perdida nitrógeno promedio por dieta:
```

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

```
## D      376.50
## C      315.75
## B      233.75
## A      195.00
## dtype: float64
```

De este modo, se ve que las dietas con mayores perdidas de nitrógeno son C y D. 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): 1014.5417
```

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

```
## Valor t-crítico: 4.3178
```

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

```
## Valor LSD con alfa=0.001: 97.2482
```

```
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: 38.7500
##   Resultado: La diferencia NO es significativa (diferencia <= LSD)
##
## - A vs. C:
##   Diferencia de medias: 120.7500
##   Resultado: La diferencia es SIGNIFICATIVA (diferencia > LSD)
##
## - A vs. D:
##   Diferencia de medias: 181.5000
##   Resultado: La diferencia es SIGNIFICATIVA (diferencia > LSD)
##
## - B vs. C:
##   Diferencia de medias: 82.0000
##   Resultado: La diferencia NO es significativa (diferencia <= LSD)
##
## - B vs. D:
##   Diferencia de medias: 142.7500
##   Resultado: La diferencia es SIGNIFICATIVA (diferencia > LSD)
##
## - C vs. D:
##   Diferencia de medias: 60.7500
##   Resultado: La diferencia NO es significativa (diferencia <= LSD)

```