

OPTIMIZANDO EL DESEMPEÑO ACADÉMICO

Equipo JJAR (Julieta, José, Alonso, Rebe)

PLANTEAMIENTO

Queremos optimizar el promedio de un semestre dado que ya pasaron bajas

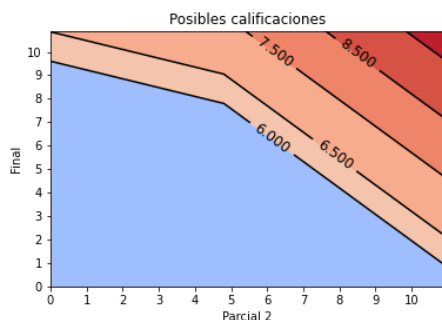


OPTIMIZANDO CALIFICACIONES PARA MAXIMIZAR LA NOTA FINAL

FUNCIÓN DE CALIFICACIÓN

Queremos encontrar la mayor calificación final posible que podríamos obtener dado que ya conocemos la calificación del 'primer parcial'.

Para no caer en un problema lineal consideramos funciones de calificación no-lineales.



O: ¿POR FIN VOY A PASAR ECO 2?

PLANTEAMIENTO

Queremos optimizar el promedio de un semestre dado que ya pasaron bajas

FUNCIÓN DE CALIFICACIÓN

Queremos encontrar la mayor calificación final posible que podríamos obtener dado que ya conocemos la calificación del 'primer parcial'.

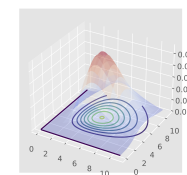
Para no caer en un problema lineal consideramos funciones de calificación no-lineales.

Por ejemplo: Peso de parciales depende de si es el más alto o bajo

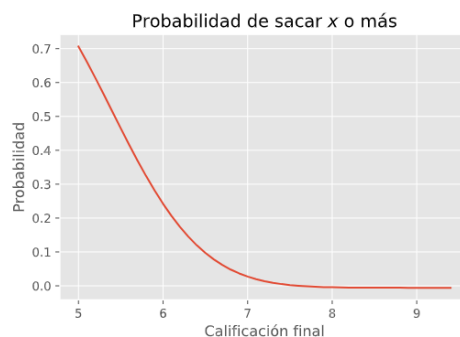
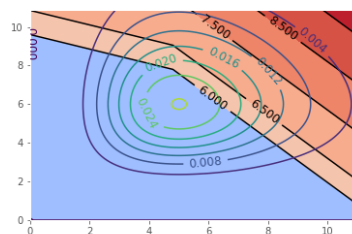
$$C(p, y) := \frac{1}{5}p_{lo} + \frac{2}{5}\left(\sum_i p_i\right) + \frac{2}{5}y$$

¿QUÉ TAN PROBABLE ES TENER X DE CALIFICACIÓN FINAL?

ASUMIENDO QUE LAS CALIFICACIONES SE DISTRIBUYEN GAMMA



JUNTANDO AMBAS IDEAS



¿CUÁL ES NUESTRA FUNCIÓN A OPTIMIZAR?

FUNCIÓN SCORE

Calcula las posibles calificaciones finales dados los datos, criterios de calificación y calificaciones dadas, ponderando por la probabilidad de obtener dicha calificación.

Variables de optimización: x, y

Concretamente:

Sean $X, Y \sim \text{Ga}(\alpha_i, \beta_i)$ v.a's independientes.

Concretamente:

Sean $X, Y \sim \text{Ga}(\alpha_i, \beta_i)$ v.a's independientes.

X representa parcial restante, Y examen final.

$$\text{Score}(x, y) := C(p, y) * f_{X,Y}(x, y)$$

Concretamente:

Sean $X, Y \sim \text{Ga}(\alpha_i, \beta_i)$ v.a's independientes.

X representa parcial restante, Y examen final.

$$\text{Score}(x, y) := C(p, y) * f_{X,Y}(x, y)$$

Donde $C(p, y)$ es la función de calificaciones y $f_{X,Y}$ la f.d.p conjunta de X, Y .

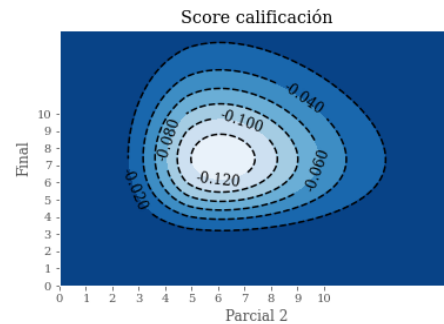
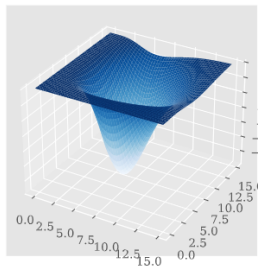
```
import numpy as np
from scipy.stats import gamma

def obj(x):
    pdf = lambda x, y: gamma.pdf(x, a=7) * gamma.pdf(y, a=8)
    return -100 * score(4.8, pdf, x[0], x[1])

x0 = [8.4, 8.4]

xk, pts = region_confianza(obj, nd.Gradient(obj),
                           nd.Hessian(obj), x0)
```

¿CÓMO SE VE SCORE?



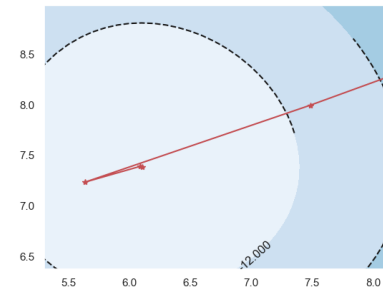
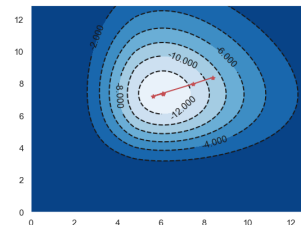
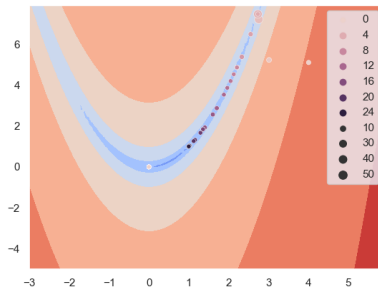
Obtenemos la máxima posible calificación \times probabilidad de sacarla.

¿CÓMO OPTIMIZAMOS?

Algoritmo de Región de Confianza

- método de Dogleg

RESULTADOS

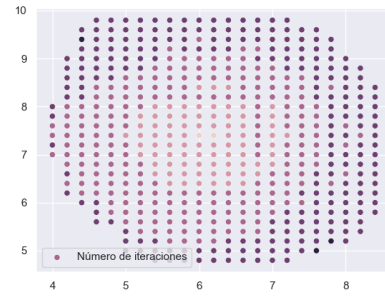


SOLUCIÓN EN (6.10, 7.38)

NO CONVERGE EN TODOS
LADOS

NO CONVERGE EN TODOS
LADOS

PROBLEMAS DE
CONVERGENCIA



- Excelente documentación
- Pruebas unitarias

- El modelo propuesto no es muy bueno

- El modelo propuesto no es muy bueno
- Posible problema de escalamiento (pg.26)

- El modelo propuesto no es muy bueno
- Posible problema de escalamiento (pg.26)
- numdifftools