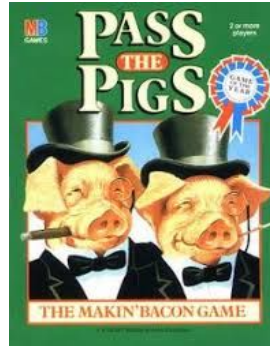


Las matemáticas de ser un cerdo avaricioso



Taller de Talento Matemático de Navarra
Nafarroako Matematika-Talentu Tailerra

Tudela, 17 de octubre de 2025

El juego de los cerditos («*Pass the Pigs*»)

Objetivos

- Aprender cómo se juega (es muy simple).
- Modelizar matemáticamente la manera óptima de jugar a *Pass the Pigs*.
- ¡Divertirnos un rato! (Y aprender matemáticas).



Introducción

¿Cómo se juega?

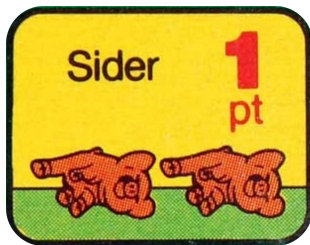
- Son cerditos de plástico.
- Tienen dibujado un puntito negro en un costado y nada en el otro.
- Lanzamos los cerditos y ganamos puntos en función de cómo quedan.

«¿Y eso es todo? No parece muy buen juego...»

Bueno, realmente no lo es... Pero sí que tienes una decisión como jugador: puedes **volver a tirar**.



Introducción



Podemos volver a tirar tantas veces como queramos en nuestro turno para ganar más puntos, salvo que los cerditos queden así:



A esta configuración se le llama «**cerdada**», en cuyo caso pierdo los puntos que haya podido ganar este turno y paso los cerdos al otro jugador.



Introducción

Así que mi única opción como jugador es «plantarme o pedir», «doble o nada», «tirar o pasar»... Como lo queramos llamar.

La única decisión que tomamos es: «¿Tiro otra vez, o... Es arriesgarse demasiado?»

A pesar de que es un juego muy sencillo, no es obvio saber cuál es la manera óptima de jugar.

Problema

¿Cuáles son las probabilidades de las tiradas?

¡Simplifiquemos el problema!



El juego

Reglas del juego «*Pig*»

- En su turno, el jugador lanza tantas veces como quiera.
- Si obtiene un 2, 3, 4, 5 o 6, gana esos puntos.
- Si obtiene un 1, pierde los puntos de la ronda y pasa el turno al otro jugador.
- Gana la primera persona que llegue a 100 puntos.

De hecho, el juego *Pig* con un dado de 6 caras era el juego original.

(Pero al parecer es más divertido lanzar cerditos de plástico).

¿Sabríais razonar cuál es la manera óptima de jugar?

¡Adelante!

El juego

- Para que quede claro, los dados **NO** están trucados.
- No podemos predecir quién va a ganar (hay un factor aleatorio).
- Lo que sí podemos tratar de predecir es qué va a ocurrir **en media**.

¿Tenéis alguna intuición de qué es lo que habría que hacer para jugar bien?

- **Opción 1.** Decidir un número n de lanzamientos y parar. ¿Cuál es el mejor n ?
- **Opción 2.** Decidir una puntuación p a alcanzar y parar. ¿Cuál es la mejor p ?

¿Cuál de las dos perspectivas es mejor?

¿Complica las cosas jugar contra alguien?

Estrategia de lanzamientos

Vamos a analizar primero la estrategia de lanzamientos. Algunos datos...

- Al lanzar el dado una vez, ¿qué probabilidad hay de obtener un 1?
- ¿Qué probabilidad hay de obtener algo que no sea 1?
- ¿Cuál sería el **valor esperado**, E , al lanzar una vez?

$$E = \frac{1}{6} \cdot 0 + \frac{1}{6} \cdot 2 + \frac{1}{6} \cdot 3 + \frac{1}{6} \cdot 4 + \frac{1}{6} \cdot 5 + \frac{1}{6} \cdot 6 = \frac{2 + 3 + 4 + 5 + 6}{6} = 3, \overline{33}$$

Otra manera de verlo:

$$\frac{1}{6} \cdot 0 + \frac{5}{6} \cdot 4 = 3, \overline{33}$$

Estrategia de lanzamientos

¿Cuál es el valor esperado al lanzar **dos veces**?

$$\frac{5}{6} \cdot \frac{5}{6} \cdot 2 \cdot 4 = 5, \overline{56}$$

¿Podemos generalizarlo para n lanzamientos?

$$E_n = \left(\frac{5}{6}\right)^n \cdot 4n$$

Estrategia de lanzamientos

Damos valores y obtenemos una gráfica:

n	E_n
1	3,33
2	5,56
3	6,94
4	7,72
5	8,03
6	8,03
7	7,81



Estrategia de puntuación

¿Cuál es vuestra intuición? ¿Qué puntuación objetivo sería la óptima para jugar?

- Está claro que si intentamos llegar a una puntuación de 10, en ocasiones obtendremos más de 10, pero otras veces sacaremos un 1 y obtendremos 0.
- ¿Cuál sería la puntuación media que obtendríamos?

¡Resulta que no es para nada intuitivo! Y es muy, muy complicado de calcular...

Podemos aproximarnos al resultado poniéndonos en un caso concreto:

Por ejemplo: supongamos que tengo 10 puntos, ¿qué ocurrirá si lanzo de nuevo?

Estrategia de puntuación

Supongamos que tenemos 10 puntos:

$$\frac{1}{6} \cdot 0 + \frac{5}{6} \cdot (4 + 10) \approx 11,67$$

- Si tengo 10 puntos y no tiro de nuevo, obtengo 10 puntos.
- Si tengo 10 puntos y tiro de nuevo, en media, obtengo 11,67 (más que 10).

Supongamos que tenemos 30 puntos:

$$\frac{1}{6} \cdot 0 + \frac{5}{6} \cdot (4 + 30) \approx 28,33$$

Si tengo 30 puntos y tiro de nuevo, en media, obtengo 28,33... ¡Menos que 30!

Estrategia de puntuación

¿Podemos lograr hallar el valor de p donde tenemos la igualdad?

Pero antes... Vamos a ver alguna simulación.

¿Alguien se anima a intentar conseguir 50 puntos en un turno?

¿Cuántas veces podría tener que intentarlo antes de conseguirlo?

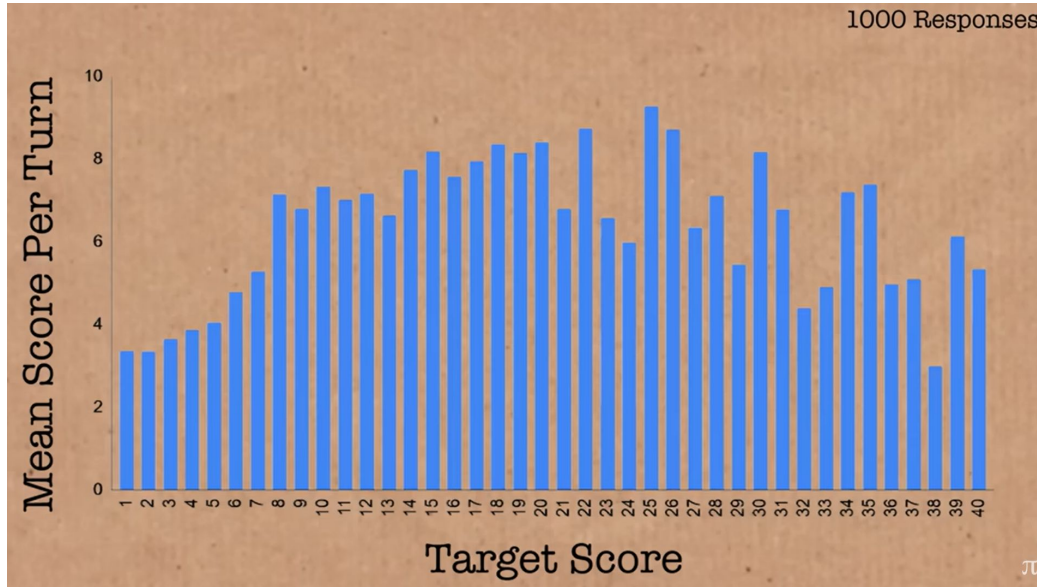
Ahora, para la (in)ecuación que buscamos...

$$\frac{5}{6} \cdot (p + 4) > p \implies 5p + 20 > 6p \implies 20 > p$$

$$p < 20$$

Datos

¿Podemos recopilar datos suficientes jugando para estimar el resultado $p < 20$?

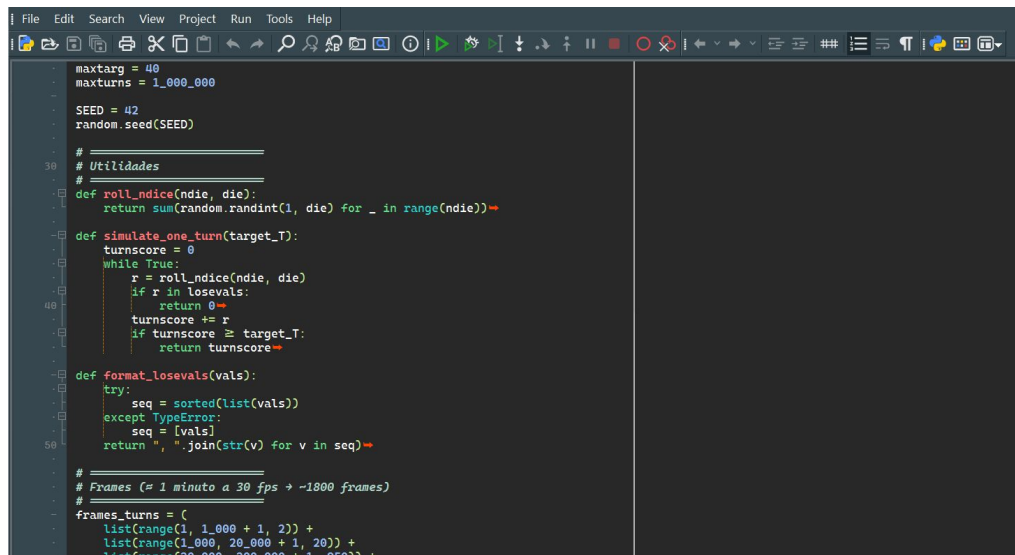


Estadística de juego basada en mil respuestas de alumnos.

Fuente: Ben Sparks (Numberphile).

Necesitamos más datos obtenidos mediante simulaciones por ordenador:

- Escribimos un programa con Python.
- Simulamos 1.000.000 de turnos con cada estrategia.
- Anotamos la puntuación media por turno con cada una.

A screenshot of a Python code editor with a dark theme. The code defines a simulation for a game. It starts with constants for 'maxtarg' (40) and 'maxturns' (1,000,000), a 'SEED' (42), and a random seed. A comment indicates 'Utilidades'. A function 'roll_ndice' takes 'ndie' and 'die' and returns the sum of random integers. Another function 'simulate_one_turn' takes 'target_T' and loops until a target score is reached, adding random dice rolls. A third function 'formatlosevals' formats a list of values. At the bottom, a comment says '# Frames (# 1 minuto a 30 fps -> ~1800 frames)' and a list 'frames_turns' is initialized with three ranges of turn numbers: 1 to 1,000 + 1, 1,000 to 20,000 + 1, and 20,000 to 700,000 + 1.

```
maxtarg = 40
maxturns = 1_000_000

SEED = 42
random.seed(SEED)

#
# Utilidades
#
def roll_ndice(ndie, die):
    return sum(random.randint(1, die) for _ in range(ndie))

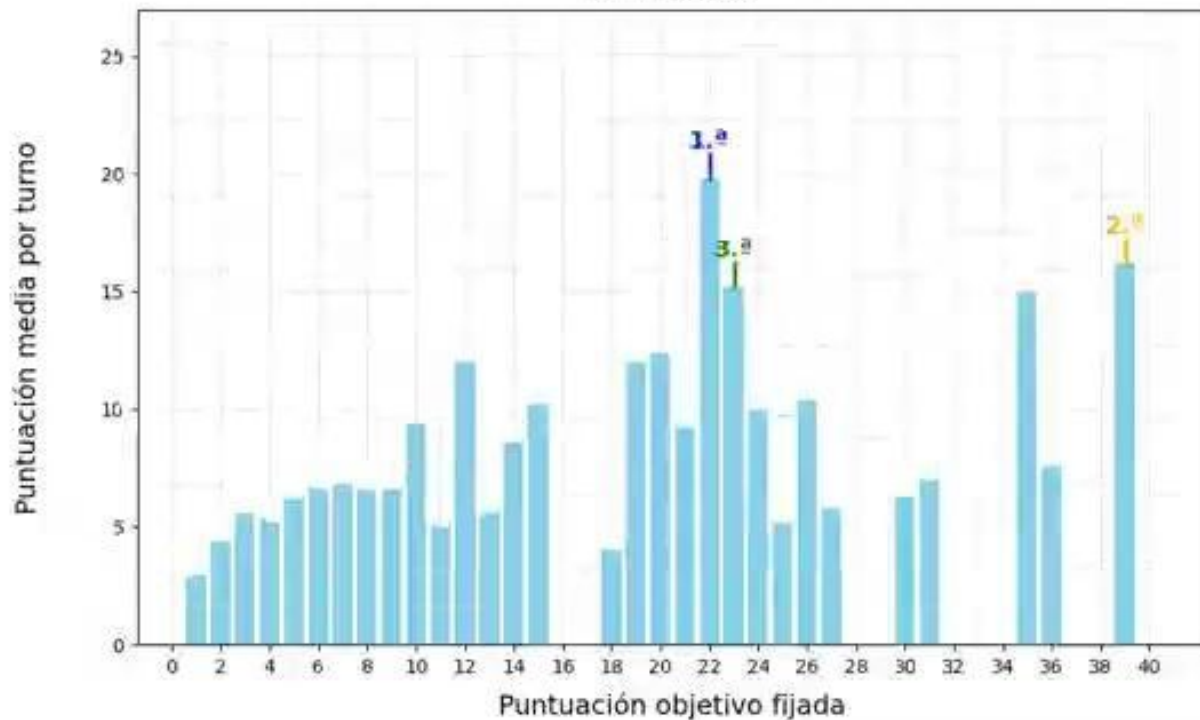
def simulate_one_turn(target_T):
    turnscore = 0
    while True:
        r = roll_ndice(ndie, die)
        if r in losevals:
            return 0
        turnscore += r
        if turnscore >= target_T:
            return turnscore

def formatlosevals(vals):
    try:
        seq = sorted(list(vals))
    except TypeError:
        seq = [vals]
    return ", ".join(str(v) for v in seq)

#
# Frames (# 1 minuto a 30 fps -> ~1800 frames)
#
frames_turns = (
    list(range(1, 1_000 + 1, 2)) +
    list(range(1_000, 20_000 + 1, 20)) +
    list(range(20_000, 700_000 + 1, 950)) +
```

Dado de 6 caras - Fin de turno al lanzar un 1

En 5 turnos.



- En los primeros turnos, no se ve ningún patrón claro.
- Después de 1.000 turnos, los datos parecen más precisos que los enviados.
- Las puntuaciones objetivo altas no lo hacen tan mal como cabría esperar.
- Las barras correspondientes a «1» y «2» tienen la misma altura. *¿Por qué?*

Conclusión: si fijamos 20 como puntuación objetivo, a la larga, maximizamos la puntuación media por turno.

¿Es esto lo que buscábamos?

- Estamos considerando las probabilidades de puntuar en nuestro turno.
- ¡La probabilidad de obtener una u otra puntuación en un turno **no** es exactamente lo mismo que la probabilidad de ganar!
- En cualquier caso, este modelo nos da una base para comenzar a razonar:
 - a. Tomamos 20 como puntuación «base» a la que aspirar.
 - b. Si vamos rezagados, debemos arriesgar un poquito más.
 - c. Si vamos por delante, tenemos margen para bajar un poco el objetivo.

El resultado anterior es un resultado «experimental», no teórico.

Si intentamos calcular de manera teórica, como en la estrategia de lanzamientos, cuál es el valor esperado cuando fijas 20 como puntuación objetivo, el problema es **más complicado**. Se trata de un problema de partición (cuántas maneras hay de descomponer 20 como suma de 2, 3, 4, 5, 6 pudiendo repetirlos).

Resulta curioso que sea más fácil hacer un 1.000.000 de partidas simuladas para estimar un valor teórico que intentar calcular dicho valor *a priori*. Este método en matemáticas es conocido como **método de Montecarlo**.



John von Neumann y Stanislaw Ulam.

Fuente: Wikipedia.

¿Cómo jugar contra otra persona?

Nuestro análisis se basa en maximizar nuestra puntuación media por turno, pero esto no es exactamente lo mismo que maximizar nuestras probabilidades de ganar.

El análisis no es sencillo. Hay tres variables a considerar:

- Puntuación del jugador 1.
- Puntuación del jugador 2.
- Puntuación objetivo (aquí sería 100).

Neller T. W. & Presser C. G. M (2004). *Optimal Play of the Dice Game Pig*.

¿Cómo jugar contra otra persona?

En su artículo, los autores parten del valor base 20 y ajustan la estrategia óptima usando una técnica conocida como «*value iteration*».

There is no known general method for solving equations of the form $\mathbf{x} = \max(\mathbf{A}_1\mathbf{x} + \mathbf{b}_1, \mathbf{A}_2\mathbf{x} + \mathbf{b}_2)$. However, we can solve our particular problem using a technique called value iteration.

Solving with Value Iteration

Value iteration [Bellman 1957; Bertsekas 1987; Sutton and Barto 1998] is a process that iteratively improves estimates of the value of being in each state until the estimates are “good enough.” For ease of explanation, we first introduce a simpler game that we have devised called “Piglet.” We then describe value iteration and show how to apply it to Piglet as a generalization of the Jacobi iterative method. Finally, we describe how to apply value iteration to Pig.

Optimal Play of the Dice Game Pig. Fuente: Numberphile.

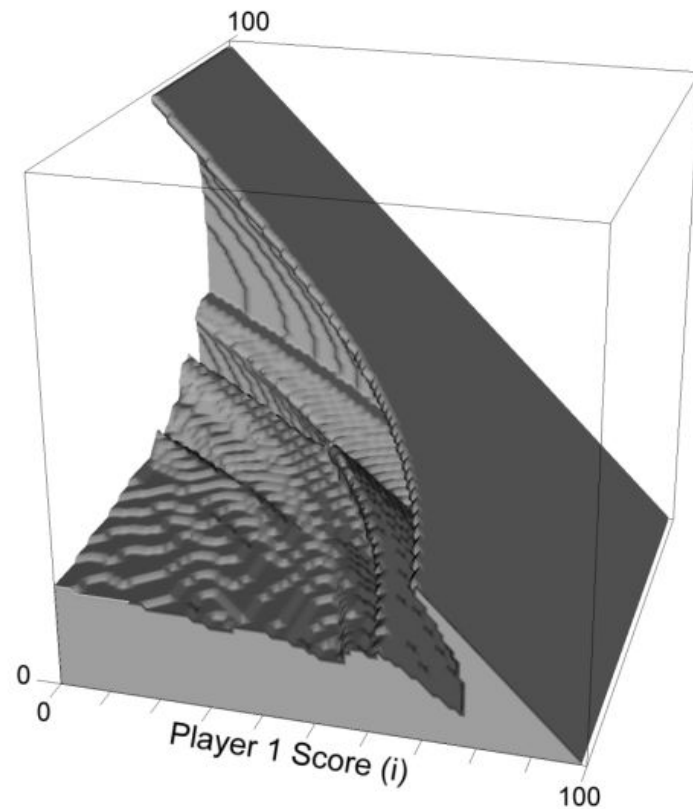
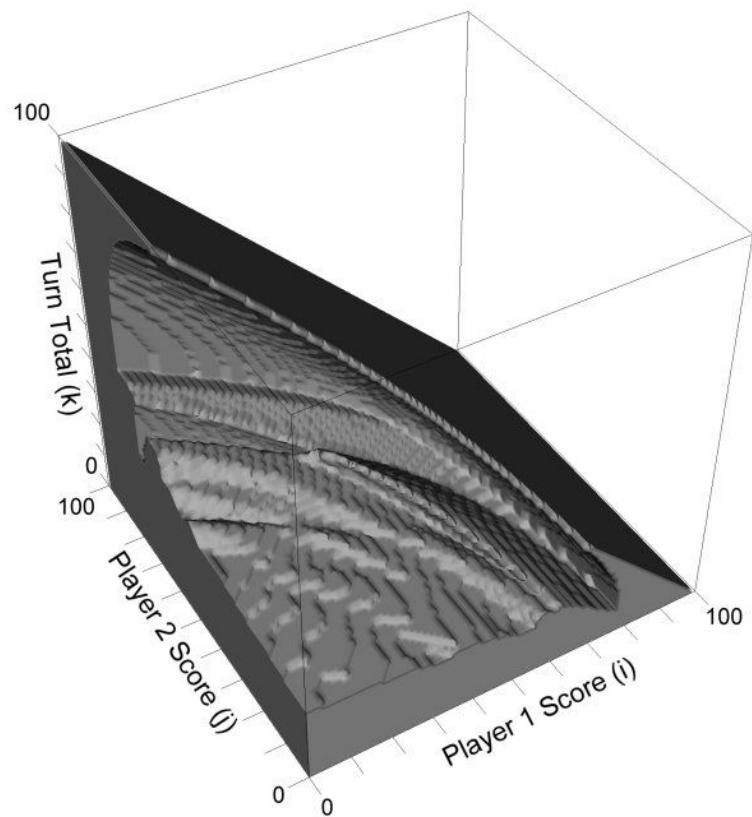
¿Cómo jugar contra otra persona?

De manera muy resumida, los autores observaron que a toda posición del juego definida por «Puntuación del J1», «Puntuación del J2», «Puntuación obtenida en la ronda» se le puede asociar un valor que básicamente describe cómo de probable es ganar el juego desde esa posición.

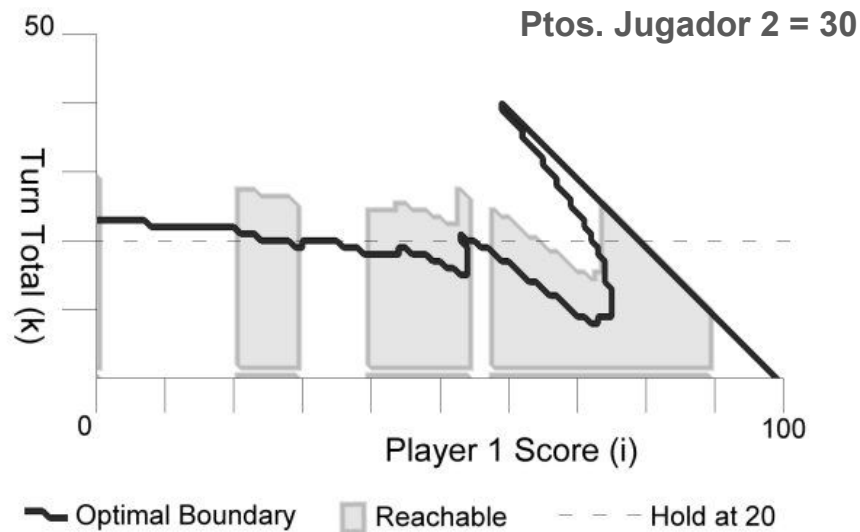
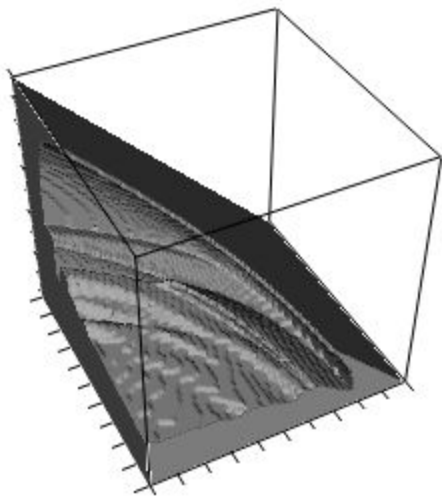
- Este valor depende esencialmente de a qué posiciones puede el jugador aspirar a llegar desde una posición dada, y cómo de valiosas son esas posiciones.
- A partir de ahí, formulan un sistema de ecuaciones enorme, asignan valores aleatorios a todas las posiciones y realizan un cálculo que mejora la estimación del valor de cada posición con cada iteración.

Finalmente, obtienen una gráfica tridimensional que indica «cómo jugar»

¿Cómo jugar contra otra persona?



¿Cómo jugar contra otra persona?



¿Cómo jugar contra otra persona?

Con su método, Neller & Presser no sólo han logrado encontrar estrategias óptimas para el juego *Pig*, sino también para *Pass the Pigs*.



65%



20%



9%



4%



1%

[Enlace al simulador online Pig](#)



¡MUCHAS GRACIAS!

¿Cuestiones, dudas, sugerencias...?

`jserrander@educacion.navarra.es`