```julia
using PlutoUI, FileIO, Images
```

```julia
using DifferentialEquations, Plots, LinearAlgebra, CalculusWithJulia, SymPy,
ForwardDiff
```

Tarea Método de Euler implicíto

Actividad realizada por Alan Acero, Johan López y Nicolás Duque

El método de Euler implícito es un esquema numérico utilizado para resolver ecuaciones diferenciales ordinarias (EDOs). A diferencia del método de Euler explícito, que avanza la solución utilizando información de la derivada en el punto actual, el método de Euler implícito utiliza la derivada en el punto siguiente.

Derivación del método

Dada una EDO de la forma:

$$\frac{dy}{dt} = f(t, y) \qquad y(t_0) = y_0,$$

integramos ambos lados en el intervalo $[t_i, t_{i+1}]$:

$$\int_{t_i}^{t_{i+1}} \frac{dy}{dt}\, dt = y(t_{i+1}) - y(t_i) = \int_{t_i}^{t_{i+1}} f(t, y(t))\, dt.$$

Despejando $y(t_{i+1})$, obtenemos:

$$y(t_{i+1}) = y(t_i) + \int_{t_i}^{t_{i+1}} f(t, y(t))\, dt.$$

Aproximamos la integral utilizando el método del rectángulo superior, donde evaluamos f en el punto final t_{i+1}:

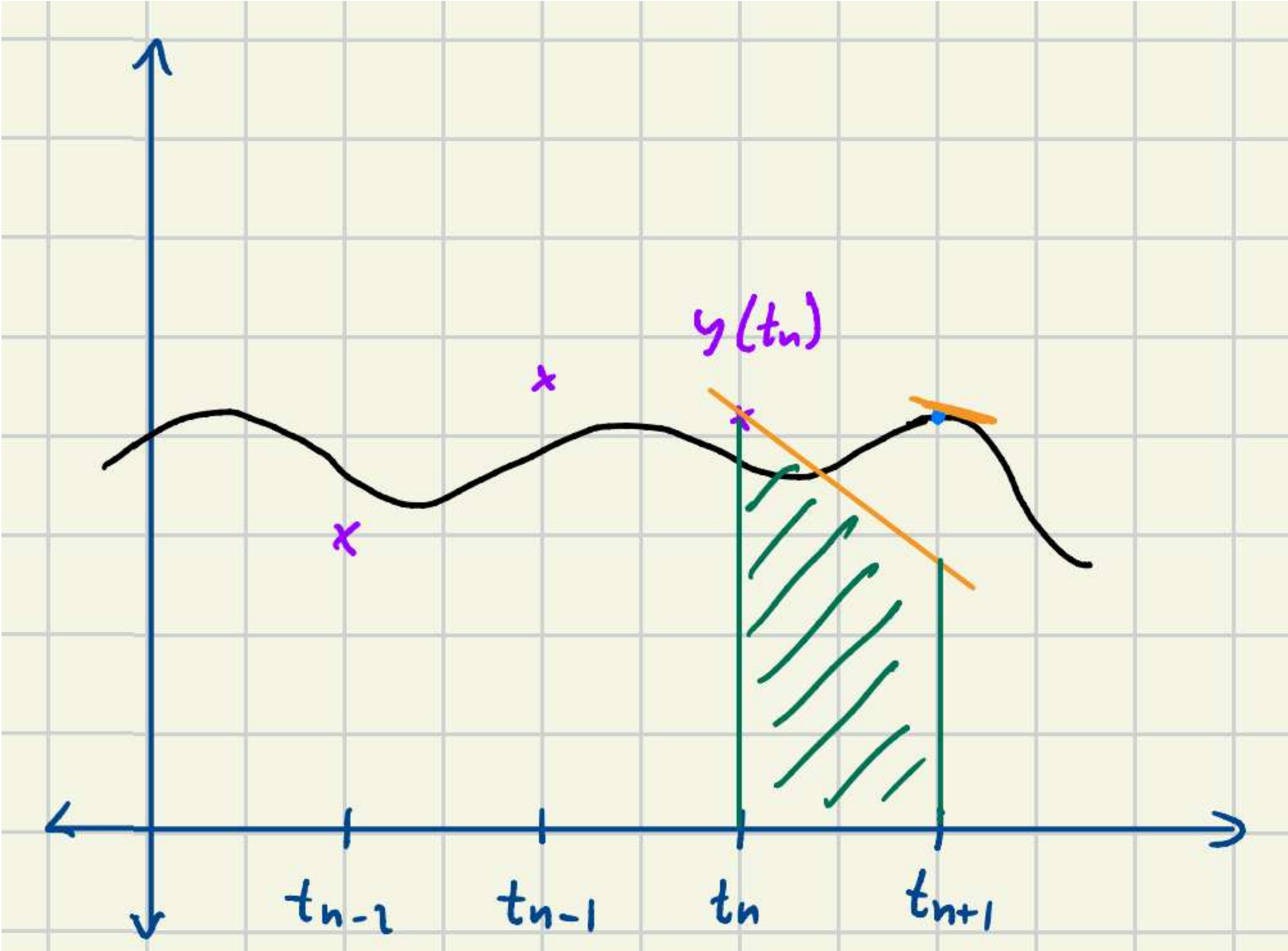
$$\int_{t_i}^{t_{i+1}} f(t, y(t))\, dt \approx hf(t_{i+1}, y(t_{i+1})).$$

Aquí, $h = t_{i+1} - t_i$ es el tamaño del paso.

Sustituyendo esta aproximación en la ecuación anterior, obtenemos la iteración del método de Euler implícito:

$$y(t_{i+1}) = y(t_i) + hf(t_{i+1}, y(t_{i+1})).$$

Esta ecuación es implícita porque $y(t_{i+1})$ aparece en ambos lados, y a diferencia del metodo anterior, acá no podremos remplazar para hallar nuestro siguiente paso, sino que tendremos que resolver la ecuación, uno de los metodos mas usados es el de Newton-Rapson.

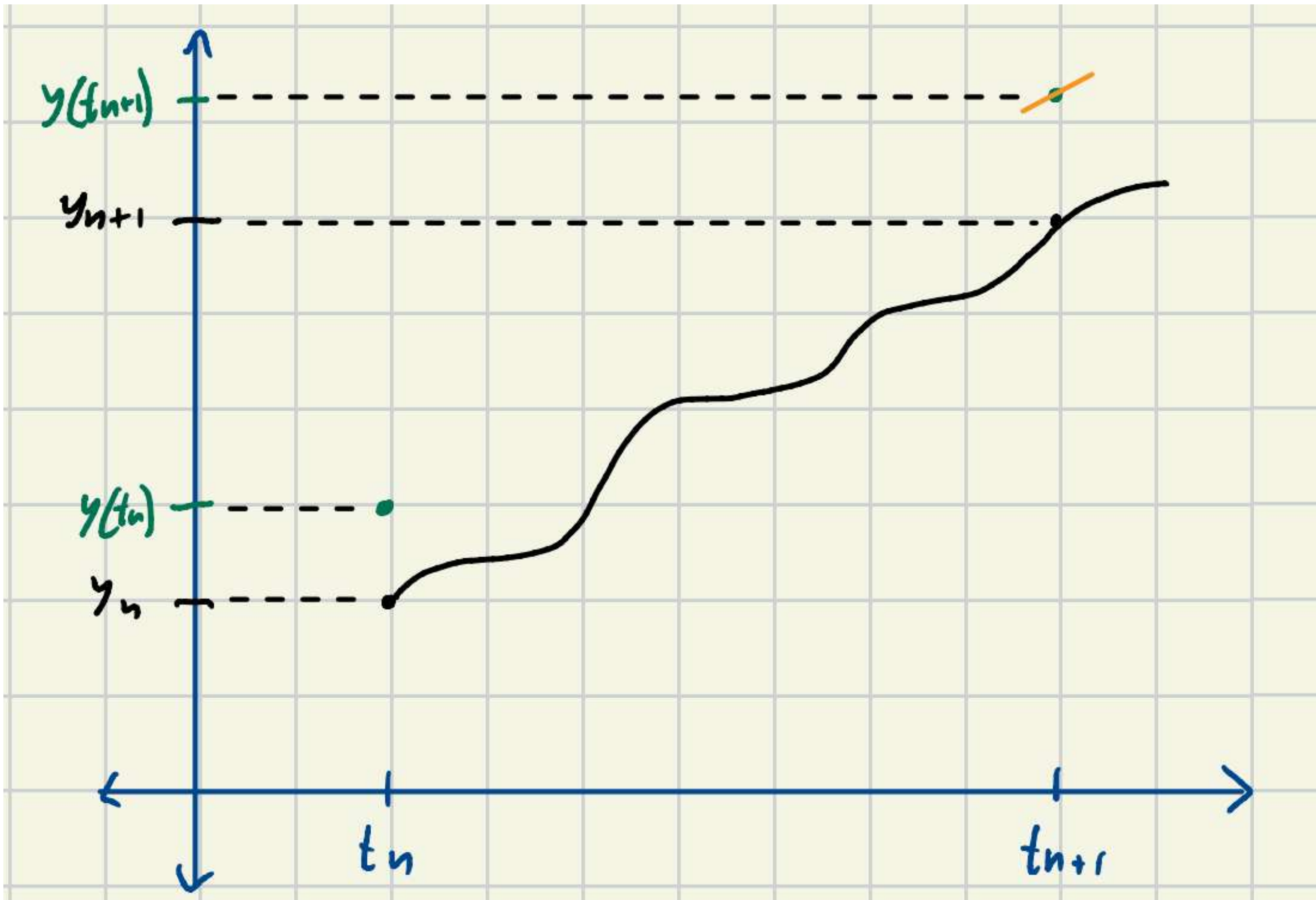


Dentro de esta imagen podemos ver una idea que se profundarizara más adelante de como la iteración va aproximando a la función

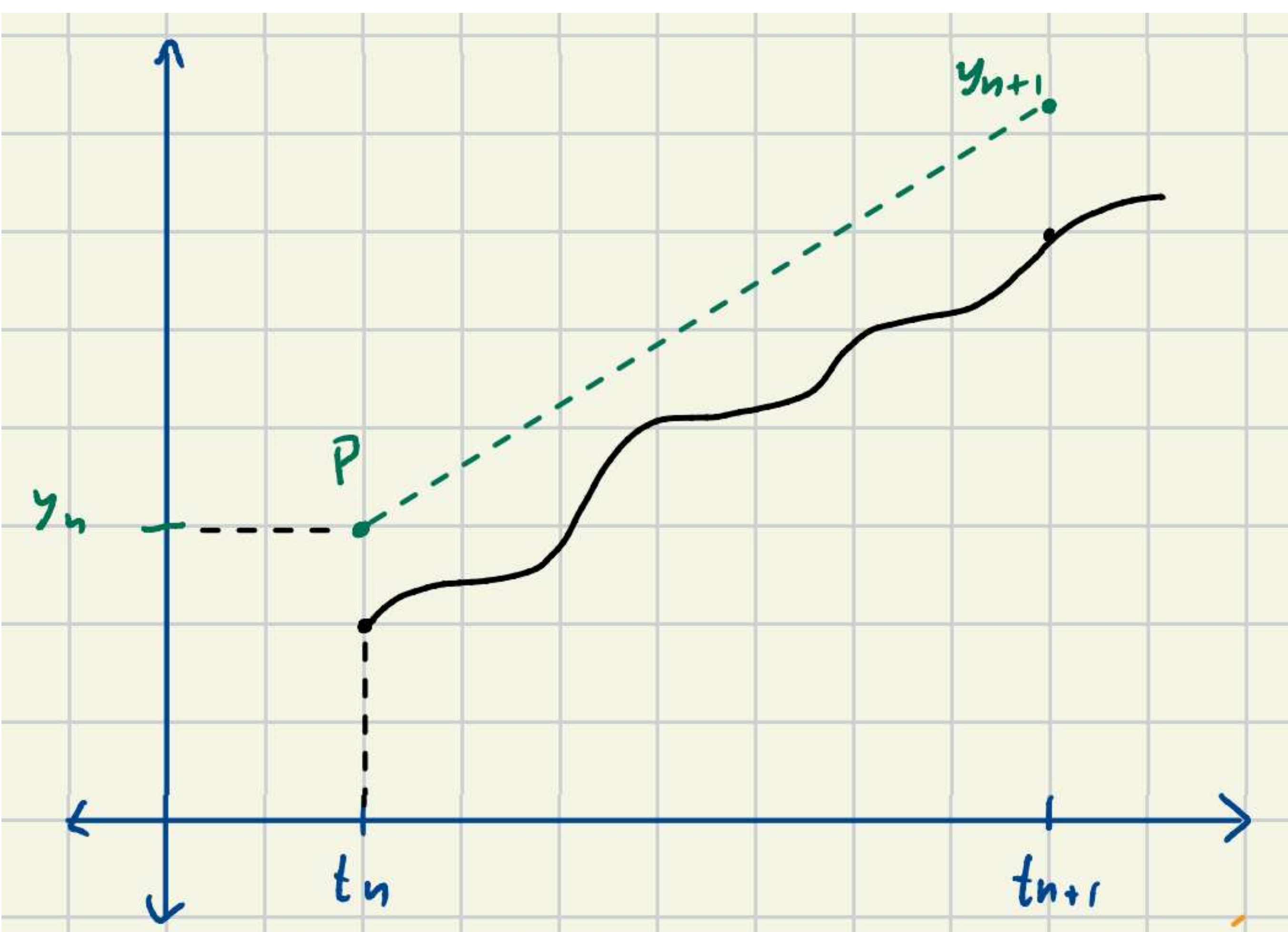
Bueno, ¿y eso que tiene que ver con las empanadas?

Titulo alterno: Interpretación gráfica

Para nuestra interpretación, consideramos la curva de la función solución exacta. En el método de Euler implícito, la pendiente se evalúa en el punto futuro y_{i+1}, lo que lo diferencia del método de Euler explícito, donde la pendiente se evalúa en el punto actual y_i.



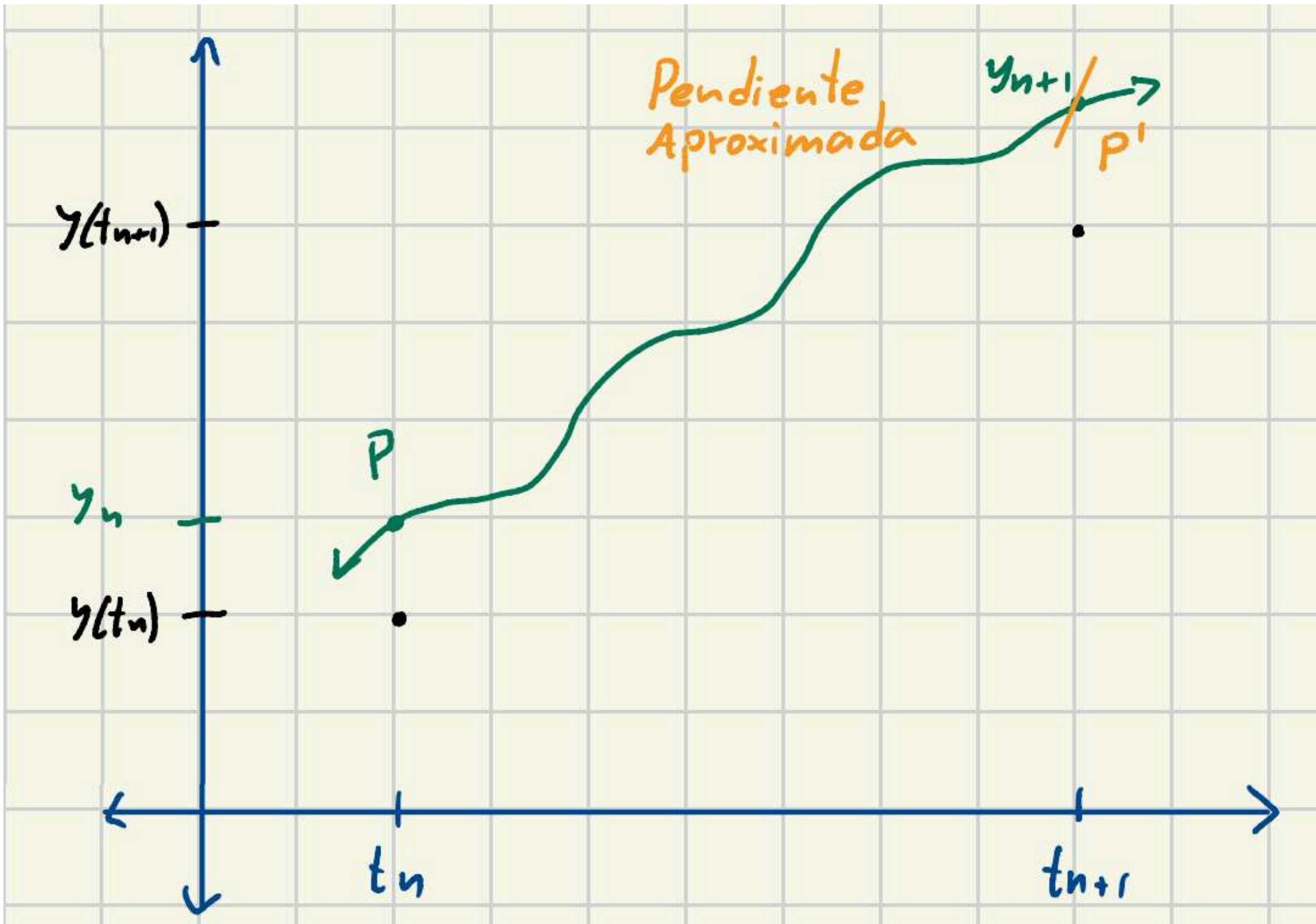
Dada nuestra curva, y nuestra pendiente en el punto t_{n+1}, lo que buscamos hacer entonces es "trasladar" esa pendiente a nuestro punto t_n, y desde ahí mandar la recta, como se muestra en la siguiente imagen



Y ¡Listo! ya tenemos nuestra interpretación grafica, ¿verdad?

no, aún no

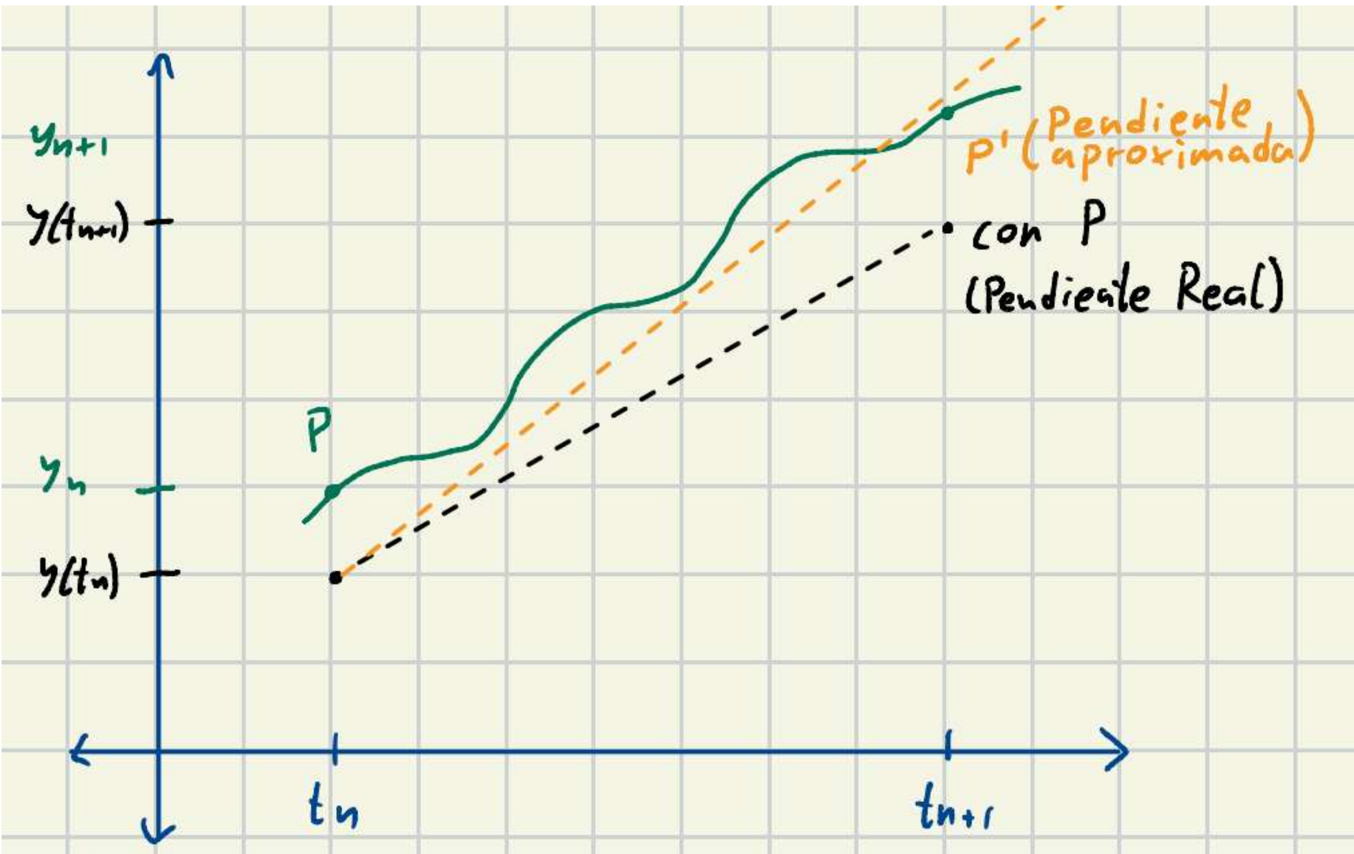
La cuestión con nuestro metodo implicito es que no tenemos el valor de $y(t_{i+1}) = y(t_i) + hf(t_{i+1}, y(t_{i+1}))$, ya que nos estamos basando en el valor que precisamente queremos encontrar, así que para cada iteración debemos de encontrar un valor a traves de metodos númericos como el de newton-rapson para el hallar el valor $i + 1\text{-esimo}$, así tenemos la siguiente grafica:




```
load("img/3.1.png")
```

dada la pendiente aproximada p', vamos a repetir la idea que teniamos de antes con la pendiente real p:

```
md"""
dada la pendiente aproximada $p'$, vamos a repetir la idea que teniamos de antes con la pendiente real $p$:
"""
```



como se puede notar, la pendiente real daria para un metodo más periodico, similar al explicito, pero al tener una pendiente alterada, puede variar respecto a lo que realmente buscamos.