

Gradiente Descendente (IIA 21/3)

Una inmobiliaria desea automatizar la tarea de tasar terrenos. El archivo `inmobiliaria.txt` contiene la base de datos de casas en Portland, Oregon. La primera columna corresponde con la dimensión del terreno (en pies cuadrados), la segunda corresponde a la cantidad de dormitorios y la tercera al precio (en dólares).

(a) *Regresión Lineal:*

1. Implementar una regresión lineal, utilizando gradiente descendente (elegir el *learning rate* con prueba y error), a partir de los datos para tasar la propiedad en función de las otras características. El código debe estar estructurado de la siguiente manera:

```
class regresion_lineal:
    # Inicializar atributos y declarar hiperparámetros
    def __init__(self,...

    # Etapa de entrenamiento
    def fit(self,X,y):

    # Etapa de testeo
    def predict(self,X):

    # Método oculto que normalice
    def _normalize(self,X,is_training: bool = True):
```

A su vez, debe poder extraer los atributos `reglin.w` y `reglin.b`.

2. Entrenar la regresión lineal con el dataset mencionado anteriormente.
3. Utilizar el regresor para predecir el costo de una propiedad de 1650 pies cuadrados y 3 dormitorios.

(b) *Learning-rate óptimo:* Agregar un método a la implementación anterior que permita calcular el *learning rate* óptimo. ¿Para que valores hay convergencia?

(c) *Comparación:* Repetir el entrenamiento para el valor hallado en el inciso anterior. Comparar los resultados de los parámetros aprendidos, la predicción mencionada y la cantidad de iteraciones.