

Aprendizaje Automático y Minería de Datos

Clasificador de Pokémons

Diego Baratto Valdivia

Leonor Cuesta Molinero

Contents

[SVM: 2](#_Toc29732102)

[Descripción del Proyecto: 2](#_Toc29732103)

[Resultados Obtenidos: 2](#_Toc29732104)

[Conclusiones: 2](#_Toc29732105)

[Regresión logística: 2](#_Toc29732106)

[Descripción del Proyecto: 2](#_Toc29732107)

[Resultados Obtenidos: 2](#_Toc29732108)

[Conclusiones: 2](#_Toc29732109)

[Redes Neuronales: 2](#_Toc29732110)

[Descripción del Proyecto: 2](#_Toc29732111)

[Clasificador de Legendarios: 2](#_Toc29732112)

[Predecir el tipo: 2](#_Toc29732113)

[Resultados Obtenidos: 2](#_Toc29732114)

[Conclusiones: 3](#_Toc29732115)

# SVM:

## Descripción del Proyecto:

## Resultados Obtenidos:

## Conclusiones:

# Regresión logística:

## Descripción del Proyecto:

## Resultados Obtenidos:

## Conclusiones:

# Redes Neuronales:

## Descripción del Proyecto:

Utilizando redes neuronales, hemos creado tanto un clasificador de legendarios en función de sus otras características como un predictor del tipo de Pokémon que es en función de su relación con otros tipos y características.

### Clasificador de Legendarios:

Tomando como referencia distintas características de entre *[attack, base\_egg\_steps, base\_happiness, base\_total, capture\_rate, defense, experience\_growth, height\_m, hp, percentage\_male, sp\_attack, sp\_defense, weight\_kg]*.

Se ha utilizado como referencia la práctica de laboratorio basada en redes neuronales, así como el módulo de Python “Keras” con “Tensorflow”.

En este clasificador, entrena una red neuronal dividida en grupos de entrenamiento, validación y testing cogiendo ejemplos del grupo total de forma aleatoria y barajándose para evitar que los legendarios y no legendarios se agrupen al principio o final de los grupos.

Para tratar de asegurar el mejor *score* de entrenamiento, se repite el proceso un número establecido de veces y guarda el mejor.

### Predecir el tipo:

## Resultados Obtenidos:

## Conclusiones: