L2F test

Submission
Professor Oak expects a 30 minutes presentation where you will have the chance to convince him to use the tool you have developed.
The presentation will take place after the challenge.
All types of resources may be used to complete the test.

Design the Pokedex 2.0

Pokémon are small creatures that fight in competitions. All Pokémon have different numerical characteristics (strength of attack, defense, etc.) and belong to one or two so-called classes (water, fire, etc.).

[Professor Oak](https://www.pojo.com/cartoon/Oak.gif) is the inventor of the [Pokedex](https://en.wikipedia.org/wiki/Gameplay_of_Pok%C3%A9mon#Pok%C3%A9dex), a useful portable device that keeps information about all the Pokémon available. As his lead data scientist, you just received a request from him asking you to update the software on all Pokedex devices. Professor Oak, who is known to be unpatient, is expecting you to present the results obtained while (1) exploring the dataset of Pokémon and the battles they fought, (2) developing a model to predict the outcome of battles between Pokémon. Even if Professor Oak loves to leave space for creativity to his lead data scientist, he gave you a list of points you might want to analyse for your presentation. The latter doesn't have to be meant as a strict list of things that Professor Oak expects from you, so feel free to follow your ideas.

IMPORTANT! Since Professor Oak uses to change his mind quickly, make sure you check your mailbox frequently till the end of the time.

Dataset description

Professor Oak has dumped the memory of one Pokedex device, resulting in the dataset you'll work with in this test.

In the file pokemon.csv, each row represents the features of one Pokémon.

* pid: Numeric - ID of the Pokémon
* HP: Numeric - Health points
* Attack: Numeric - Strength of the regular attack
* Defense: Numeric - Strength of the regular defense
* Sp. Atk: Numeric - Strength of the special attack
* Sp. Def: Numeric - Strength of the special defense
* Speed: Numeric - Moving speed
* Legendary: Boolean - 'True' if the Pokémon is rare
* Class 1: Categorical - Pokémon class
* Class 2: Categorical - Pokémon class

Please note that a Pokémon can have either one or two classes. If a Pokémon has two classes, they are both considered to have the same importance.

In the file combats.csv, each row represents the outcome of one battle between two Pokémon.

* First_pokemon: Numeric - ID (match with pid)

* Second_pokemon: Numeric - ID (match with pid)

* Winner: Numeric - ID of the winner

Part 1: Exploring the data

1. Explore the data and report some descriptive statistics (e.g., use df.describe(), report the number of classes, summarize the attack and defense distributions, etc.).

2. Compare the probability distribution of the “regular attack” feature with that of the “regular defense” feature. In particular:

   1. visualize the relation between these two variables using an appropriate plot;

   2. list the names of the 3 Pokémon with highest attack-over-defense ratio;

   3. list the names of the 3 Pokémon with lowest attack-over-defense ratio;

   4. list the names of the 10 Pokémon with the largest number of victories.

3. Professor Oak suspects that Pokémon in the _*grass*_ class have a stronger regular attack than those in the _*rock*_ class. Check if he is right and convince him of your conclusion with statistical arguments.
4. Having in mind that the final goal is to build a model to predict the output of battles between Pokémon, generate the feature vectors and the labels to train your model, forming the training set.
5. Explore the training set.

Part 2: Prediction

The model should take as input the features of two Pokémon and generate a binary value to predict who of the two will win.
1. Create features based on the knowledge you gained of the data, and add them to the training set.

2. Train several models to predict the winner of a match based on the available features. To do so, use appropriate validation approaches.

3. Summarize and describe the results obtained with different models, highlighting the differences.
4. Find your best model(s). Motivate your choice.

5. Analyse feature importance of your best model(s).

Good luck!