Decision Trees for exploratory analysis of Legendary-Pokémon and their predictors

Pascal Quindeau
University of Niederrhein
Krefeld, Germany
quindeau.pascal@gmail.com

Name Surname
name of organization (of Aff.)
City, Country
email address

***Abstract*—In this paper, we first scrape relevant Pokémon data from two distinct websites and further use it to explore different distributions among Pokémon labelled as either *Non-Legendary*, *Sub-Legendary*, *Mythical* and *Legendary*. We identify several possible predictors that might give a Pokémon its *Legendary* status, such as its *Stats*, its first and second *Type*, its *Growth Rate* and even its *Height*. Finally, we use a decision tree model to get more insight into how these multiple predictors function together to distinct between *Legendary* and *Non-Legendary* Pokémon. Our decision tree models achieves an *accuracy score* of 0.964 which is roughly 6% better than the baseline model.**

***Index Terms*—Data Science, Decision Tree, Pokémon, Exploratory Analysis**

I. Introduction

In recent years, Data Science has emerged as an important discipline for business companies, individuals and even society [1]. Van der Aalst refers to it as an "amalgamation of classical disciplines like statistics, data mining, data bases and distributed systems". In this paper, we are mostly interested in data mining and statistics. For the sole data mining purpose, Fayadd et al. discuss the need of Data Mining as a discipline of knowledge discovery in databases (short KDD) in [2] as data in its raw format is mostly abstract. A statistician, data scientist or analyst has first to transform the data into some more compact and useful form. Here, a data science project usually follows a given pattern:

1) Selection: Select the needed data from a larger data storage. This could be a file, a database or even the web.
2) Preprocessing: Transform the data, for example from unstructured to structured or introduce new features as a combination of other features.
3) Transformation: Transform the data into a valid format that is helpful for the analysis task.
4) Data Mining: Use specific Data Mining tools, like exploratory models, descriptive statistics and more.
5) Interpretation/ Evaluation: Conclude your findings and add them as new or confirmed knowledge.

For this paper, we decided to mirror the aforementioned approach. Here, we use Data Mining and Data Science interchangeable as we are mostly interested in extracting new knowledge from our data and not how to distribute it or store it efficiently in a database. Our data domain will be Nintendo's most liked Pokémon-Franchise[1]. Pokémon is a well known and well understood phenomenon among young people and even adults. The data is publicly available on several websites and can be easily scraped via Tools like scrapy[2].

This paper will be structured as follows: First, we talk about the data collecting and preprocessing step. For this, we scrape two websites, one for the general Pokémon data and one for the target variable. Next, we provide some exploratory inside into the data itself and form multiple hypothesis based on distribution plots and pivot charts. At last, we train a decision tree model on a subsample of the overall available data and report its accuracy as well as the exploratory insight that a decision tree provides.

II. Data preparation

A. Scraping the data

The data is collected from two different website. The general Pokémon Database[3] provides a list of all Pokémon sorted by their national Dex number. This is further known as a *National Pokédex*. A *Dex* number is a unique identifier for each Pokémon. Currently, it is made up of a three digit integer preceded by a #. Due to its uniqueness it can be used as a primary key if stored in a database. The list can be easily scraped using the Python framework scrapy. For each Pokémon, scrapy will follow the response to the Pokémon's actual web page, that we further use to extract meaningful data. Table I gives an overview of the scraped features. Note that some of these might be missing for several Pokémon. A Pokémon might not have a second *Type* as it is single typed. Some Pokémon like "Zarude" were only introduced recently and are not available to the game yet. Thus, data is missing. Another problem with using Pokémon Database lies in the way, the website stores special evolutions like *Mega Evolutions*. These are listed under the same name and *Dex* entry and cannot be extracted for a given Pokémon without overwriting it's first occurrence in a database. Thus, we decided to not include these evolutions in our database.

To extract label information, such as *Non-Legendary*, *Legendary*, *Sub-Legendary* and *Mythical*, we use Serebii.net[4].

[1] See https://www.pokemon.com/de/ for more information about Pokémon.
[2] https://scrapy.org/
[3] https://pokemondb.net/pokedex/
[4] https://www.serebii.net/pokemon/legendary.shtml

TABLE I: Overview of extracted data from the Pokémon Database.

Feature	Description
Dex Number	A Pokémon's unique identifier.
Name	A Pokémon's name.
First and second Type	A Pokémon's typing. Categorical value.
Height	A Pokémon's height. Numerical value.
Weight	A Pokémon's weight. Numerical value.
Stats	A Pokémon's statistics. Including Attack, Defense, Special Attack, Special Defense, Speed, Health Points and Total, whereas Total is the sum of all stats. Numerical values.
Catch Rate	The rate of successfully catching a Pokémon. It reaches from 0 (low) to 255 (high). Numerical value.
Growth Rate	The rate determining how much experience a Pokémon needs to reach a certain level. Can be one of the following: Erratic, Fast, Medium Fast, Medium Slow, Slow, Fluctuating. Categorical value.
Male	Indicating whether a Pokémon can be male. Binary value.
Female	Indicating whether a Pokémon can be female. Binary value.
Male Rate	The probability of a Pokémon being male. Numerical value.
Female Rate	The probability of a Pokémon being female. Numerical value.
Gen	The Gen in which a Pokémon first was introduced.

This site lists all Pokémon categorized by the aforementioned labels. We then use a Pokémon's *Dex* number to combine both datasets. In the end, we extract 893 Pokémon. The overall distribution of labels among different Pokémon can be seen in table II. It shows a high skew towards *Non-Legendary* Pokémon. Around 90% of Pokémon are *Non-Legendary*. We have to keep this information in mind, when training our decision tree classifier later.

III. Exploratory analysis

In this section, we explore several possibilities for predictors that could help to distinct *Legendary* from *Non-Legendary* Pokémon. We do this by looking at different kinds of plots and descriptive values among the different labels. Based on the finding, we can then formulate specific hypothesises. This is not common in statistics, where scientists usually first formulate a hypothesis and try to proof it by analysing the given data. In this paper, however, we just want to give a short overview about basic exploratory techniques.

A. Stats Distribution

First, we want to investigate a Pokémon's stats (short for statistics). Here, we use a very versatile seaborn[5] function, called *pairplot*. A pairplot shows the interaction between a set of variables on a matrix grid. Each cell corresponds two one combination. Combinations of different variables show scatter plots, whereas combinations of same variables show a distribution in form of a histogram or kernel density estimation (kde). Figure 1 shows the aforementioned pairplot for all Pokémon and their stats. It is notable, that most *Non-Legendary* Pokémon lie in the bottom left of each given plot. Moreover, there seems to be a correlation between Hp and Atk as well as Spd and Satk. Indeed the correlation happens to be 0.46 and 0.42, respectively. The different distributions among the stats are most prominent in the offensive stats (Atk, Satk and Spd). Especially *Legendary* Pokémon tend to differ strongly from *Non-Legendary* Pokémon. This effect is more subtle for *Sub-Legendary* Pokémon. This leads to our first hypothesis:

1) *Non-Legendary* Pokémon tend to have lower stats, than other Pokémon.

It might be, that the difference among stats is only due to non evolved Pokémon in the *Non-Legendary* group. Usually, *Legendary*, *Mythical* and *Sub-Legendary* already have reached their final form, whereas *Non-Legendary* Pokémon may go through two or three stage in their life. That is, why one might observe a bimodal distribution among *Non-Legendary* Pokémon. This, however, is not the case.

[5]https://seaborn.pydata.org/

TABLE II: Distribution of Pokémon among labels.

Label	Count/Frequency
Non-Legendary	806/0.902
Sub-Legendary	42/0.047
Mythical	23/0.026
Legendary	22/0.025

B. Height and weight distribution

Next, we report our findings on a Pokémon's height and weight. Especially *Legendary* Pokémon tend to be large and ominous looking beasts. Pokémon like Kyogre[6] and Groudon[7] come to mind. Kyogre weights 352.0 Kg with a height of 4.5m where Groudon is a real heavy weight with 950 Kg and 3.5m. The dataset contains two Pokémon with a maximal weight of 999.9 Kg. These are Cosmoem and Celesteela. The largest Pokémon, however, is the new *Legendary* Pokémon called Eternatus with a height of 20.0m.
Figure 2 shows the height and weight distribution as two boxplots. At first glance, we observe the large difference in height between *Legendary* and other Pokémon. This shows, that *Legendary* Pokémon indeed are larger than other Pokémon.

[6]https://www.pokewiki.de/Kyogre
[7]https://www.pokewiki.de/Groudon

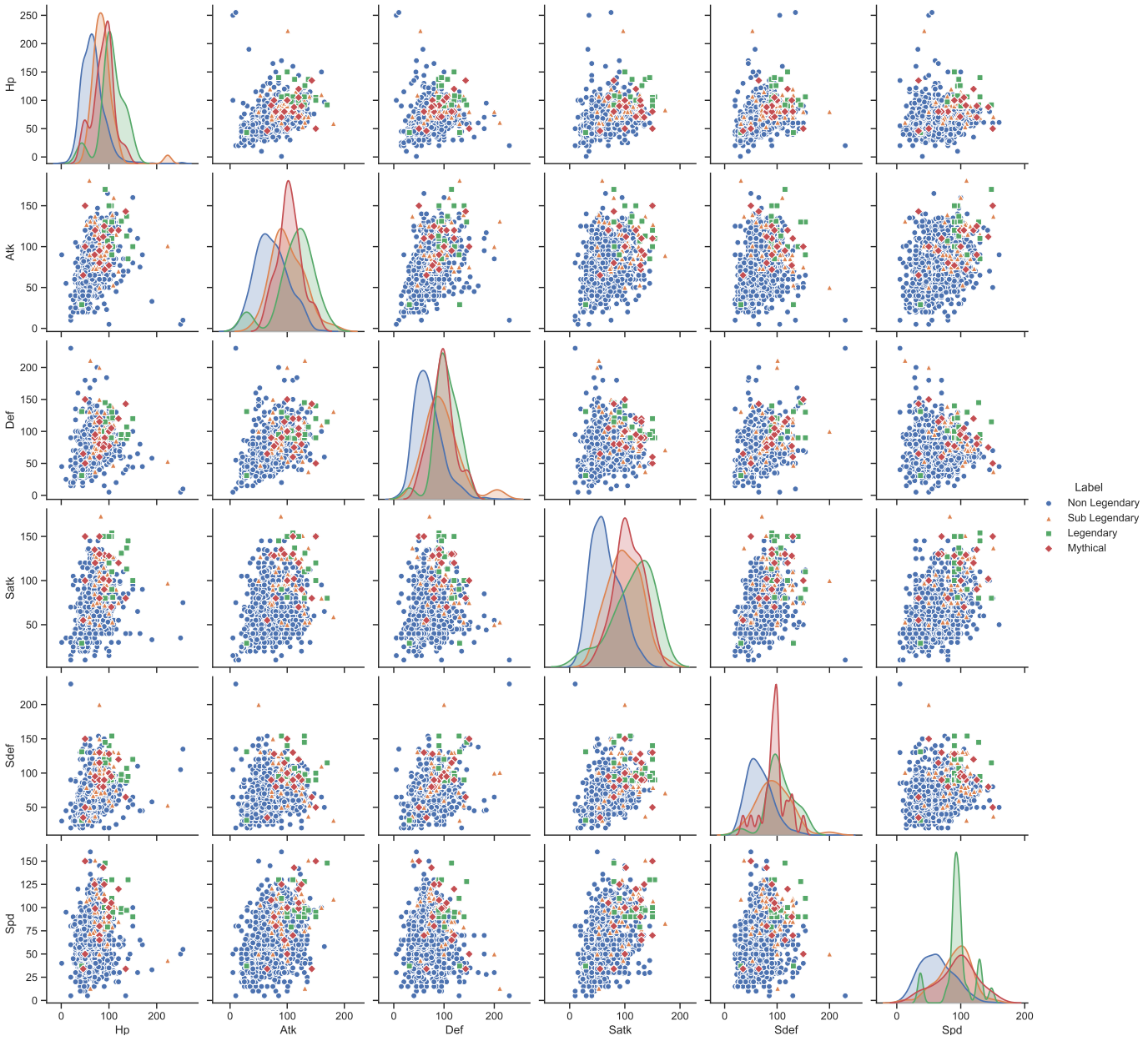
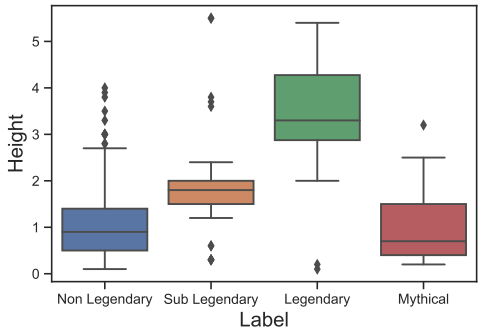


Fig. 1: Pairplot of Health Points (HP), Attack (Atk), Defense (Def), Special Attack (Satk), Special Defense (Sdef) and Speed (Spd). The labels are colour and shape coded.

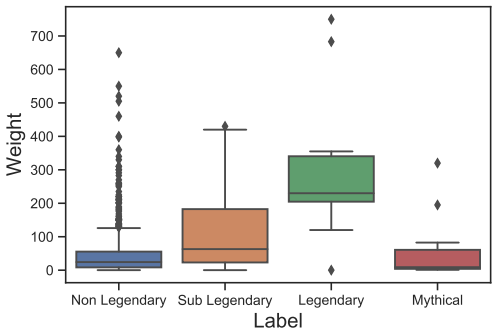
This effect is not as pronounced on *Sub-Legendary* Pokémon, where 75% lie between 1.5 to 2.0 metres. In section **??**, we talk about the influence on a Pokémon's first and second Type. A lot of *Legendary* Pokémon tend to be either Dragon or Psychic Pokémon. In another boxplot, that groups the data by first Type instead of Label, it shows, that Dragon Types tend to be 0.5m larger in average than other Pokémon. This finding shows, that the height advantage of *Legendary* Pokémon might be due to the frequent Dragon Type.

Another interesting finding relates to *Mythical* Pokémon. Notice, how most *Mythical* Pokémon weight almost nothing. In fact, 50% weight only 0.1Kg or slightly more. This may be important, when combining all three Labels into a generic *Legendary* definition. The weight distribution might be bimodal with extreme values on both ends corresponding to *Generic Legendary* Pokémon. We conclude this subsection with three more hypothesises:

2) *Legendary* Pokémon are larger than other Pokémon.
3) *Legendary* Pokémon weight more than other Pokémon.
4) When merging *Legendary*, *Sub-Legendary* and *Mythical* Pokémon to a Group called *Generic Legendary* Pokémon, extreme light weight Pokémon tend to belong to that class as well.



(a) Height



(b) Weight

Fig. 2: Boxplot of the height and weight distribution among the different Pokémon classes.

C. Genders among Pokémon

In this section, we want to highlight the importance of a Pokémon's Gender for the given classification task. Despite most animals in biology following a binary gender-distribution, Pokémon actually allow for a third gender, that is having no gender at all. To include this in our analysis, we create a new feature called *Genderless* by combining the features *Male* and *Female* as

$$genderless_i = \neg(male_i \wedge female_i) \tag{1}$$

for all i Pokémon, where $\wedge$ defines the logical *AND*-Operator and $\neg$ the *NOT*-Operator. A one then indicates a Pokémon having no gender.

We compare the *Genderless* distribution using simple barplots for the four different labels.

As shown by figure 3 all *Legendary* and *Mythical* Pokémon are genderless, whereas only 80% of *Sub-Legendary* Pokémon show the same property. The frequency of *Non-Legendary* Pokémon, however, is nearly not noticeable. Roughly 10% of these Pokémon have no gender. This leads us to another hypothesis:

5) *Legendary*, *Mythical* and *Sub-Legendary* Pokémon show no gender at all.

Furthermore, we can use this information to implement a simple classifier. For this we summarize specific class labels

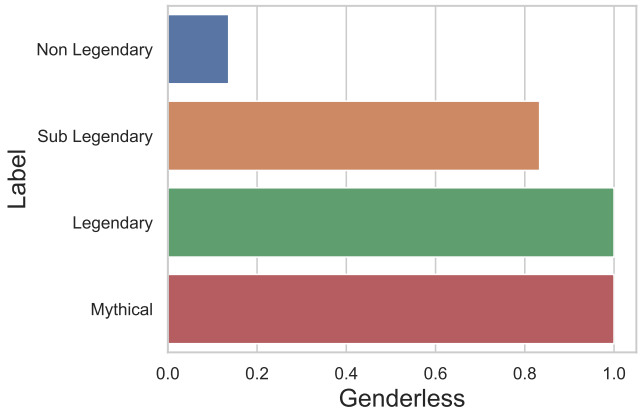


Fig. 3: Barplot of the *Genderless* distribution among different Pokémon labels, sorted by the relative frequency of Pokémon having no gender.

and aggregate them to a generic class label, simply called *Legendary*. Each Pokémon that is either *Legendary*, *Sub-Legendary* or *Mythical* will be assigned to the class 1. *Non-Legendary* Pokémon will be assigned to class 0. Then, we construct a classifier as follows

$$predict(x_{genderless}) = \begin{cases} 1 & \texttt{if } x_{genderless} = 1 \\ 0 & \texttt{else} \end{cases} \tag{2}$$

where $x_{genderless}$ is 1 for a given Pokémon if it has indeed no gender. Using this classifier on all Pokémon reaches an *accuracy* score of 0.869. This is one minus the proportion of genderless Pokémon among the *Non-Legendary* Pokémon falsely predicted as 1. This classifier, however, does worse than the simple baseline model, that predicts ever Pokémon as *Non-Legendary* (see section II-A)

D. Legendary Typing

In this section, we highlight the importance of a Pokémon's first and second *Type* regarding the given classification task. Based on our data, a Pokémon can have one of 18 Types[8]. More than half the Pokémon in a given category also have a second *Type*. The exact distribution is shown in table III. Only 21.7% of *Legendary* Pokémon have no second *Type*. 78.3%, however, have a second *Type*. This allows for interesting Type-combinations among all Pokémon, of which some are more

[8]For a list of available Types see: https://bulbapedia.bulbagarden.net/wiki/Type.

TABLE III: Shows the distribution of None Types, that is Pokémon with no second Type among the different classes of Pokémon in percentages.

Class	None Type-Frequency
Legendary	0.217
Sub-Legendary	0.409
Mythical	0.483
Non-Legendary	0.405

TABLE IV: Contingency table of Label and Type1 (Type2) among all Pokémon shown as relative frequencies of all Pokémon sharing the same Label. Highlighted are the most frequent types. Only columns are shown, where the relative frequency in one cell is equal or greater than 0.05, thus only showing important Types.

Type1 Label	Bug	Dark	Dragon	Electric	Fairy	Fire	Grass	Normal	Psychic	Rock	Steel	Water
Legendary	0.00	0.04	**0.22**	0.00	0.09	0.04	0.00	0.00	**0.30**	0.00	0.04	0.09
Mythical	0.05	0.09	0.00	0.05	0.00	0.05	0.05	0.09	**0.23**	0.05	**0.18**	0.14
Non Legendary	0.09	0.04	0.03	0.05	0.02	0.06	0.10	**0.13**	0.05	0.06	0.03	**0.14**
Sub Legendary	0.05	0.02	0.05	**0.12**	0.00	0.10	0.07	0.07	**0.12**	0.10	0.07	0.05

Type2 Label	Dragon	Fairy	Fighting	Fire	Flying	Ghost	Grass	None	Psychic	Steel
Legendary	**0.17**	0.00	0.04	0.09	**0.17**	0.04	0.00	**0.22**	0.00	0.13
Mythical	0.00	0.09	0.05	0.05	0.05	0.09	0.09	**0.41**	0.09	0.05
Non Legendary	0.03	0.04	0.02	0.02	**0.11**	0.02	0.02	**0.48**	0.04	0.03
Sub Legendary	0.05	0.10	0.12	0.00	**0.14**	0.02	0.00	**0.40**	0.05	0.07

likely as others. Our research shows that the combination of (Grass, Poison), for example, is much more likely than (Grass, Rock). The most common type combination is (Flying, Normal), where almost 35% of *Flying*-Pokémon are *Normal* as well.

Though, this finding does not help for the given classification task. Instead, we focus on the Type distribution among different classes of Pokémon. For this we use contingency tables for the two categorical values (Class, Type), where a Type can further be divided into First and Second Type.

The results are shown in table IV. Here, a cell entry indicates the relative frequency of, for example, all *Legendary* Pokémon having the Type *Bug* as a first Type, second Type respectively. The table shows, that one third of *Legendary* Pokémon has *Psychic* as their first Type, followed by the *Dragon*-Type. In comparison, only 5% of *Non-Legendary* Pokémon are *Psychic* and 3% are *Dragon*. This indicates, that belonging to a specific first or second Type can be an important factor to predict a Pokémon's *Legendary* status. This effect is most prominent with the *Psychic* Type as it shows to be a frequent Type among all Pokémon but *Non-Legendary* Pokémon.

Looking at the table we can also tell, that *Legendary* Pokémon never have *Psychic* as their second Type. This is not true for *Dragon*, though. Among the different classes of Pokémon, roughly 40% have no second Type at all. Again, *Legendary* Pokémon seem to be an exception, where only 22% do not have a second Type. This gives rise to the question, whether *Legendary* Pokémon get unique Type combinations. This might be true. Using contingency tales, we create a single table, combining the three categorical variables *Label, Type1* and *Type2* and again look at the relative frequency. Almost all Type combinations among *Legendary* Pokémon show a frequency of roughly 4.5%. Considering that only 22 Pokémon are *Legendary*, this makes us believe, that unique Type combinations are especially important for these Pokémon. Of course, with more Pokémon sharing the same Label, this leads to an increased probability, that two Pokémon share the same Type combination. With 18 Types and the possibility of no second Type the number of unique Type combinations is

$$typecomb = \frac{19!}{(19-2)! \cdot 2!} = \frac{19 \cdot 18}{2} = 171 \tag{3}$$

For 806 *Non-Legendary* Pokémon and an uniform distribution of type combinations this would mean, that about 4.7 *Non-Legendary* Pokémon would share the same Type combination or 0.58%, which is almost six times larger than $\frac{1}{806}$ or 0.12%. We conclude this exploration with one last hypothesis:

6) A Pokémon's Type is an important indicator to predict, whether it belongs to the *Legendary* or *Non-Legendary* class.

IV. Decision Trees

In the last section, we gave an exploratory insight into the difference among Pokémon belonging to different classes. We now want to give a quick theoretical overview on *Decision Trees*, the primary exploratory model, we use in for the given classification task.

Decision Trees are an effective and popular classification method initially developed by Leo Breiman et al. [3] in 1984. Bruce P., Bruce A. and Gedeck define a Decision Tree in [4, p. 250] as a "set of 'if-then-else' rules that are easy to understand and implement". For them one of the main advantages of decision trees lies in the easy explain-ability. In addition, in contrast to linear and logistic regression a decision tree is able to discover hidden patterns corresponding to complex interactions between predictors and the target variable in the data. A decision tree thus mimics the human decision making progress. Looking at loan data, a node in a decision tree may look at the borrower score of a given sample. If that score reaches a certain threshold, one may agree to the loan. When it is below the threshold, the loan is declined. This simple process can be visualized as a tree, where nodes stand for a decision and leafs for a final sub-division of the data into (hopefully) pure subsets.

In [5], Aurélien defines the impurity (1 - purity) of node i as

$$G_i = 1 - \sum_{k=1}^{n} {p_{i,k}}^2 \tag{4}$$

where $p_{i,k}$ is the ratio of class k instances among all instances in the i^{th} node and n the number of all classes. This means, when one node exists only of one class $k = 1$, then

$$p_{i,k} = \begin{cases} 1 & \texttt{if} \quad k = 1 \\ 0 & \texttt{else} \end{cases} \tag{5}$$

The impurity this equals 0, which is the best score. To achieve this, the decision tree algorithm will use the threshold for each predictor to reduce the impurity score as much as possible. Once the algorithm cannot improve the impurity significantly any more, the node becomes a leaf node. A decision tree with an impurity score of 0 in each node would be a perfect decision tree.

Unfortunately, decision trees are prone to overfitting[9]. A decision tree will adapt itself to the training data and thus perform worse on unseen or new data. One way to prevent this is using random forests instead [6]. Because random forest combine hundreds of randomized decision models in a bagging (Bootstrap Aggregation) approach, they loose some of the exploratory advantages of simpler decision trees. Instead we use *Hyperparameter Regularization*.

Decision trees come with a lot of hyperparameters to prevent an overfit. To discriminate between *Generic Legendary* and *Non-Legendary* Pokémon we finetune the following hyperparameters with a 3-fold cross validation, using a gridsearch in sklearn:

- Maximum Depth
- Maximum number of Features
- Maximum number of Leaf-Nodes
- Minimum Samples a Leaf must contain

The decision tree model is evaluated for 6300 candidates and fitted 18900 times. We keep the model with the best accuracy score. One might also consider the F1-measure, as it prevents the model from ignoring infrequent classes. We did not encounter this problem, though.

V. Results

VI. Summary

References

[1] Van Der Aalst, Wil: *"Data science in action."* Process mining. Springer, Berlin, Heidelberg, 2016. pp. 3-23.

[2] Fayyad, Usama M., Gregory Piatetsky-Shapiro, und Padhraic Smyth: *"Knowledge Discovery and Data Mining: Towards a Unifying Framework."* KDD. Vol. 96. 1996.

[3] Breiman, Leo, et al.: *"Classification and regression trees"*. CRC press, 1984.

[4] Bruce, Peter, Andrew Bruce, and Peter Gedeck: "*Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python*". O'Reilly Media, 2020.

[5] Géron, Aurélien: "*Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems"*. O'Reilly Media, 2019.

[6] Liaw, Andy, and Matthew Wiener: *"Classification and regression by randomForest."* R news 2.3 (2002): 18-22.

[9] See [5, pp. 184] for a more detailed explanation.