POKÉ-LEARNING

Sistemas de Gerenciamento de Banco de Dados

Henrique Reis Kops [*], Carlo Gallichio Laitano [†], Frederico Haigert Iepsen[‡]
Escola Politécnica — PUCRS

June 2019

1 Introdução

O aprendizado de máquina pode ser definido como uma série de regras e reajustes matemáticos/estatísticos necessários para que se encontre, ao final de uma etapa de "treino" sobre a experiência da máquina, um modelo preditivo que melhor se adeque aos dados disponíveis.

Este relatório tem como objetivo demonstar os passos e decisões tomados para a contrução de um *modelo preditivo* capaz de argumentar sobre a probabilidade de um *pokémon* [1] **ser lendário**. Dessa forma, pode-se dizer que a máquina irá aprender sobre alguns dados de diversos pokémons e terá a capacidade de avaliá-los de acordo com o rótulo já definido.

2 O dataset

Para que seja possível prever sobre a "*raridade*" de um pokémon, é necessário entender quais características (ou *features*) o classificam como tal. Em outras palavras, são necessários **dados**. Para este ecperimento, foi utilizado um conjunto de dados (comumente chamado de *dataset*) disponível pela comunidade *Kaggle*, onde se encontram algumas informações interessantes sobre 721 dessas criaturas, tais como as verificadas abaixo.

Feature	Tipo de variável	Descrição
#	Numérico	ID do pokémon.
Name	Categórico	Nome do pokémon.
Type 1	Categórico	Tipo primário do pokémon que determina fraqueza ou resistência à ataques.
Type 2	Categórico	Tipo secundário, presente em alguns pokémons.
Total	Numérico	Somatório de todas características abaixo.
HP	Numérico	Pontuação de vida ou quantidade máxima de dano que o pokémon pode receber.
Attack	Numérico	Modificador base para ataques "normais" (poder de movimento físico).
Defense	Numérico	Resistência base contra ataques "normais".
SP Atk	Numérico	Modificador base para ataques "especiais".
SP Def	Numérico	Resistência base contra ataques "especiais".
Speed	Numérico	Fator determinante sobre a velocidade de ataque (prioridade de ataque);
Generation	Numérico	Geração da franquia à qual o pokémon pertence.
Legendary	Categórico	Rotulação sobre a classificação do pokémon quanto à sua raridade (**característica alvo**).

Tabela 1: Features presentes no dataset.

[*] henrique.kops@acad.pucrs.br
[†] carlo.laitano@acad.pucrs.br
[‡] frederico.iepsen@acad.pucrs.br

[1] Criaturas fictícias resultantes do trabalho de *Satoshi Tajiri* para a franquia de jogos *Pokémon*© publicado em 1995 pela empresa *Nintendo Co. Ltd.*

2.1 Análise exploratória

Uma das etapas mais importantes para que uma máquina consiga aprender corretamente é a análise sobre os dados que serão aprendidos. Explorando-os, é possível encontrar aspectos interessantes (e "tangíveis") para que um modelo possa ser definitivamente implementado. Ainda é possível detectar se o possuído é capaz de responder a alguma pergunta previamente estabelecida.

Por motivos educacionais, esse conceito é aplicado de forma lúdica em um contexto fictício, porém, boa parte das vezes, modelos preditivos são muito úteis quando desejamos responder perguntas de negócio em grandes organizações. É importante ressaltar que é somente possível tomar uma abordagem eficaz se os dados são *organizados*, ou, então, terão de ser em algum momento.

2.2 Entendendo os dados

Como estratégia inicial de "ataque", foi aplicada uma matriz de correlação nos dados "crus", permitindo uma visualização geral sobre quais features estão mais relacionadas com a coluna alvo escolhida. Dessa forma, se obteve o gráfico demonstrado a seguir.

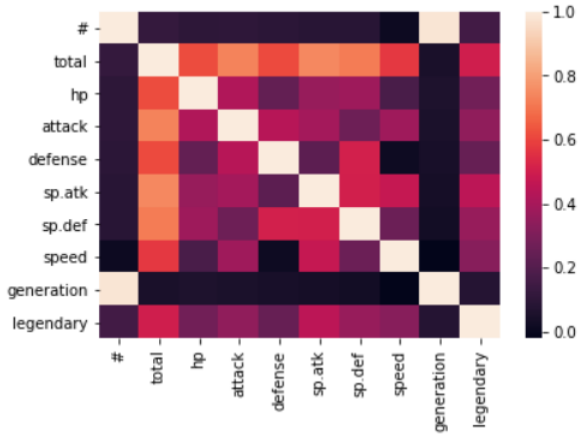

Figura 1: Captação de correlações nos dados crus.

Note que a correlação da geração e ID do pokémon não "conversam" com as outras características, portanto, como uma etapa de limpeza de dados, essas colunas foram removidas, resultando na correlação abaixo.

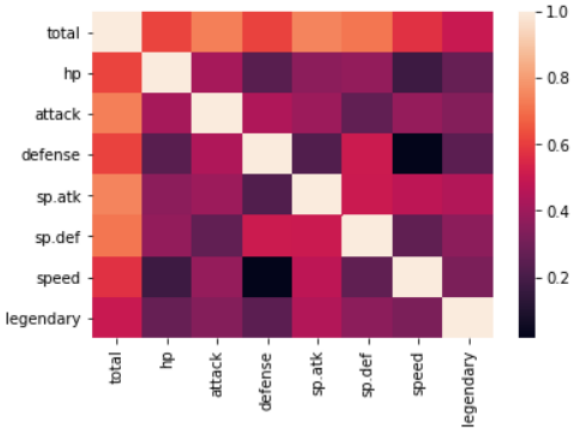

Figura 2: Gráfico de correlação após a redução de informações do dataset.

É importante que o leitor saiba sobre as restrições do classificador de correlações *Seaborn*, utilizado para gerar as visualizações anteriores. Caso se observe com cuidado, é possível notar uma diferença no número de características presentes no dataset (1) para os dois gráficos (1 e 2). Isso se deve pelo fato de que este avaliador lida somente com valores numéricos, portanto é necessário que se faça uma *discretização* das características categóricas. O resultado desse processamento se encontra abaixo.

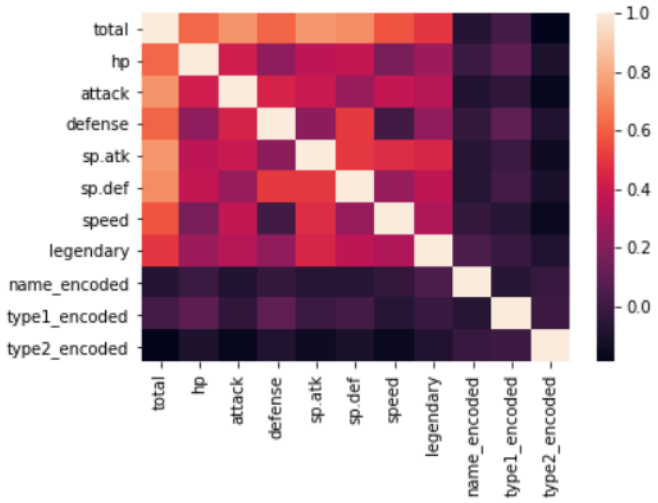

Figura 3: Caption

Novamente pode-se perceber que existem features pouco relacionadas com o resto dos dados, portanto pouco relacionadas com a característica alvo **legendary**. Tendo estas condições em mente, optou-se por utilizar os dados em seu formato mais "vivo", representado pela visualização anterior à esta (2).

3 Modelos supervisionados

A partir deste momento será iniciada a etapa de *treinamento* da máquina. Isso significa que, a partir de uma exeperiência sobre os dados, a máquina será instruída a entender sobre essa experiência com a finalidade de aprimorar aquilo que aprendeu. Assim, esta poderá prever, no contexto deste relatório, qual a probabilidade de um pokémon ser lendário. Uma das formas de se obter esse resultado é através de *modelos preditivos supervisionados*, ou seja, quando já se possui uma rotulação prévia do que será previsto. Com a finalidade de ilustrar essa idéia, seguem abaixo 5 *instâncias* (ou linhas) exemplo retiradas do dataset estudado.

#	Name	Type 1	Type 2	Total	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Generation	Legendary
244	Entei	Fire		580	115	115	85	90	75	100	2	True
245	Suicune	Water		580	100	75	115	90	115	85	2	True
246	Larvitar	Rock	Ground	300	50	64	50	45	50	41	2	False
247	Pupitar	Rock	Ground	410	70	84	70	65	70	51	2	False
248	Tyranitar	Rock	Dark	600	100	134	110	95	100	61	2	False

Figura 4: Ilustração da pré-rotulação das instâncias de treino.

Note que a coluna alvo já está rotulada por valores *booleanos*, ou seja, já estão indicados os pokémons que são lendários.

3.1 Tipos de modelos

O aprendizado de máquina se dá sob modelos preditivos. Normalmente, quando se deseja eficácia e segurança, utilizam-se modelos já existentes. Para o estudo relatado neste documento, foram utilizados os modelos supervisionados de **K-NN** (*k-near neighbours*) e **Árvore de Decisão** (*Decision Tree*).

A abordagem utilizada pelos dois algoritmos é um pouco distinta. A aplicação do K-NN busca definir uma instância *não vista* anteriormente no treinamento através da sua *distância euclidiana* para outras instâncias já treinadas, ou seja, a sua menor *dissimilaridade* no contexto em que se encontra. Dessa forma, é necessário que se tenha cuidado sobre as escalas utilizadas em características numéricas, o que não é um grande problema para este dataset.

Em uma Árvore de Decisão o fator decisivo se encontra ao percorrer um caminho nas ramificações de uma *árvore*, gerada com a finalidade de, em suas *folhas* (fim do caminho), possuir a classificação mais pura possível, ou seja, com o menor fator de "dúvida"(essa métrica é comumente calculada através do *Gini* de cada nova folha). O termo árvore é dado a um *grafo* que possui formato semelhante a uma árvore genealógica. A cada "geração", decide-se qual caminho a instância vai seguir até que esta seja completamente classificada.

Uma das grandes diferenças entre estes dois modelos é que em uma previsão por K-NN é necessário "voltar" o novo dado para todas as instâncias já classificadas, já Árvore de Decisão representa um *modelo* quando é gerada, ou seja, não existem processamentos custosos para realizar uma busca nos dados. Essa diferença faz com que os preditores baseados em K-NN sejam classificados como "*Lazy models*" (modelos preguiçosos).

3.2 Comparações

Considerando os modelos escolhidos, ambos demonstraram se encaixar muito bem no problema de classificação. Para afirmar tal sentença, é interessante argumentar sobre algumas das métricas geradas pela etapa de treino de cada um. Dessa forma, pode-se avaliar seus comportamentos através de uma técnica de validação cruzada (ou *cross validation*), utilizando diferentes *folds* (ou partições) dos dados. Isso se deve pela boa prática de aleatorização dos dados, expondo os modelos à situações "desconfortáveis", ou seja, torna-se difícil distinguir as intâncias já previstas. Essa validação grante, por exemplo, a captura de comportamentos "viciados" do modelo (*overfitting*). As métricas geradas e consideradas satisfatórias se encontram a seguir.

	precision	recall	f1-score	support
False	0.98	0.97	0.97	735
True	0.67	0.74	0.70	65
accuracy			0.95	800
macro avg	0.82	0.85	0.84	800
weighted avg	0.95	0.95	0.95	800

Figura 5: Métricas para o **K-NN**.

	precision	recall	f1-score	support
False	0.97	0.99	0.98	244
True	0.81	0.65	0.72	20
accuracy			0.96	264
macro avg	0.89	0.82	0.85	264
weighted avg	0.96	0.96	0.96	264

Figura 6: Métricas para a **Árvore de Decisão**.

4 Conclusões

A partir dos relatórios de classificação ilustrados anteriormente é notável a qualidade de reconhecimento, cuja é bem superior às pontuações de "chute" (consideradas em torno de 50%, porém pode haver variações ± x% de acordo com a hipótese nula utilizada). Sabendo que no presente dataset somente 8% dos dados rotulados estavam de acordo com a condição exposta pelo dado alvo (Legendary = *True*), ambos algoritmos pareceram bem congluentes e eficazes para fins de classificação sobre novos dados. Porém, este fator também torna válida a reflexão de que os modelos podem não ser excelentes, pois por esta baixa taxa de dados positivos torna possível a possibilidade dos modelos estarem sofrendo um *bias indutivo* sob a rotulação negativa, portanto é necessário avaliar suas outras métricas. Dessa forma, por mais que o modelo de Árvore de Decisão tenha obtido um alcance de 96% de acurácia e se sobresaiu com com 1% a menos de taxa de erro, o mesmo demonstrou um comportamento mais inadequado do que o K-NN, pois este último diverge menos quando observamos suas métricas inversas de precisão e recall. Portanto, pode-se concluir que o algoritmo baseado em K-NN se manteve mais estável, porém ambos estão longe da excelência preditiva.