



AI Education Matters: Data Science and Machine Learning with Magic: The Gathering

Todd W. Neller (Gettysburg College; tneller@gettysburg.edu)

DOI: [10.1145/3340470.3340474](https://doi.org/10.1145/3340470.3340474)

Introduction

In this column, we briefly describe a rich dataset with many opportunities for interesting data science and machine learning assignments and research projects, we take up a simple question, and we offer code illustrating use of the dataset in pursuit of answers to the question.

Magic: The Gathering (MTG) is a collectible card game featuring imperfect information, chance, complex hand management, and fascinating metagame economics in the online market for cards. One can study the in-game economy that concerns different colored/uncolored *mana*, or one can study the time-series dollar costs of cards on the open market as they are introduced and then fluctuate in supply and demand, e.g. when [rotating out of standard play legality](#) (generally after less than two years), or when found to have powerful uses in newer play formats like [Pauper](#).

As of 25 February 2019, a freely-available, rich [JSON card dataset](#) for 19386 MTG cards may be downloaded from the [MTGJSON website](#). For this column, we select a single focus question: How much more is the in-game cost for a creature with “flying” ability.

The Cost of Flying

[Flying](#) is an evasive ability of creatures that generally comes with a higher converted mana cost (CMC), i.e. the total number of in-game currency units needed to bring the card into play regardless of the specific mana colors required.

In related work, Henning Hasemann, a Berlin-based researcher and software engineer, has done preliminary data science and machine learning investigation on the cost of flying and other keyword attributes of MTG cards¹. He

Copyright © 2019 by the author(s).

¹<http://leetless.de/tag-MTG.html>

concluded that, for the same power and toughness, one pays approximately 0.79 in CMC for flying. We will take a slightly different approach in this column.

Having downloaded [AllCards.json](#), one can easily load the card data in Python using Python’s native `json` library:

```
import json
with open('AllCards.json', 'r',
          encoding='utf8') as read_file:
    data = json.load(read_file)
print(len(data), 'cards read.')
```

One can then iterate through cards and their respective data dictionaries:

```
for card_name in data:
    card_data = data[card_name]
```

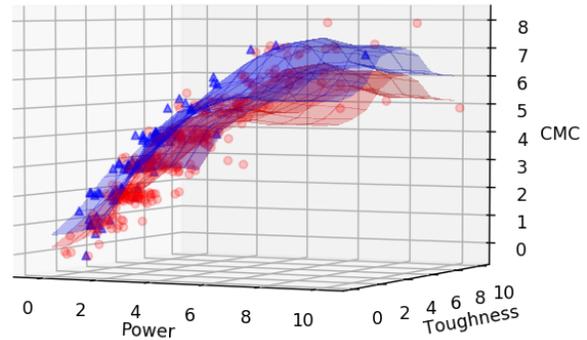
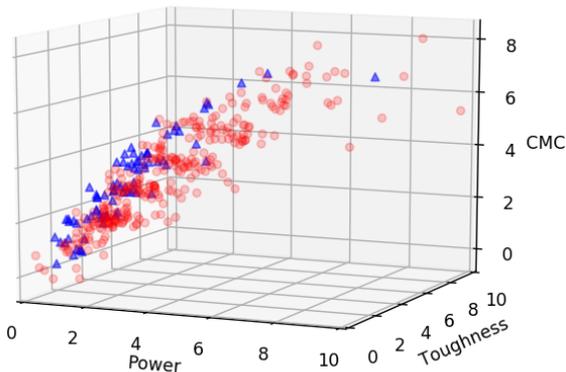
For each card dictionary `card_data`, we can access rule text, power, toughness, and CMC through dictionary keys `'text'`, `'power'`, `'toughness'`, and `'convertedManaCost'`. The structure of the JSON data is [well-documented](#).

More difficult to ascertain is which cards are creatures which are inherently “flying”. Some non-creature cards contain rule text affecting flying creatures. Other creature cards describe defense against flying creatures. A simple approach to eliminate false positive and false negatives is to filter for only creatures that have either no rule text or rule text consisting of “Flying” only. Looking over the data filtered thus, we noted that cards with “transform” layout should be excluded as their true cost of flying is the mana cost of the initial side plus the game condition that must be achieved to flip the card to its transformed side.

Instructors and students can jumpstart their exploration of the following overview by downloading [all data and code](#), including the aforementioned [AllCards.json data](#), [Python code](#), and a corresponding [Jupyter notebook](#).

In both the code and the Jupyter notebook, one can trace a simple exploration of this

question where, having filtered the 9844 creatures conservatively to include only 81 flying and 312 non-flying creatures, we first view the data as a jittered scatterplot where flying and non-flying creatures are represented as blue triangles and red circles, respectively:

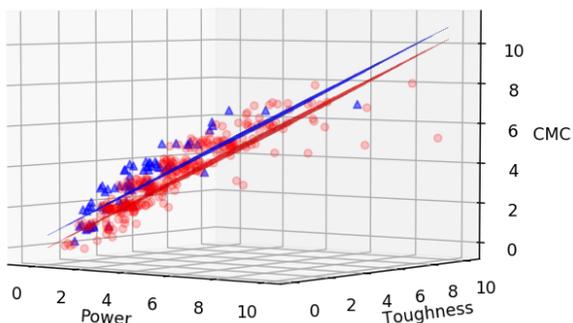


A sample-weighted mean of the flying premium with this model is about 0.725 CMC, and one can observe that the flying premium increases to about 1 for toughness values 4 and greater which lead to more difficult creature removal for the opponent.

Starting simply, we perform and visualize a linear regression predicting CMC from power, toughness, and a binary “flying” attribute, yielding a linear model with an R-squared value of 0.82:

$$\begin{aligned} \text{CMC} &= 0.6111629366158998 * \text{power} \\ &+ 0.41265041398361274 * \text{toughness} \\ &+ 0.60820253825056 * \text{flying} \\ &+ 0.2782416745722158 \end{aligned}$$

Thus, this linear model would predict a flying premium of about 0.61 CMC. Here we visualize the predictions for flying and non-flying creatures as blue and red planes, respectively:



Looking along the planes, we see non-linearity in the data. Applying gradient boosted decision tree learning (sklearn’s GradientBoostingRegressor), we see a more nuanced prediction:

However, the greater takeaway here is that this free dataset is rich and complex as the game, inviting simple inquiries like this and offering opportunities for much more complex analyses. The example code offered here is an invitation for educators and their students to take up a wide variety of fun and interesting questions concerning this popular 25-year-old game.

Past and Future MTG Research

At present, published AI research on MTG is limited to little more than a few papers on Monte Carlo Tree Search application to a simplified MTG card set (Ward & Cowling, 2009; Cowling, Ward, & Powley, 2012) and procedural MTG card generation (Summerville & Mateas, 2016). Hearthstone, a very similar digital collectible card game, has received much more research attention.

MTG should be of great future interest as a general game play challenge for a variety of reasons beyond its popularity and 25-year history. First and foremost is the greater strategic complexity arising from the rules of over nineteen thousand MTG cards. The software engineering challenge of modeling MTG is likely the reason more attention has been given to the newer and simpler game Hearthstone. An important building-block challenge here would be to design AI systems that could learn to play two fixed decks optimally against one another.

A second reason for interest is the fascinating metagame choices where players build decks for competition at time of play (e.g. through [drafting](#)) or a priori (e.g. through constructed [formats](#)) that seek to find beneficial [synergy](#) between a constrained set of allowable cards. A clever recent construction showed MTG to be Turing complete ([Churchill, Biderman, & Herrick, 2019](#)).

A third reason could be termed the *metametagame*, where the MTG consumer negotiates the supply and demand cost dynamics controlled by Wizards of the Coast LLC (WotC) by creating new play formats (e.g. [Pauper](#), [Cube](#)) to constrain costs. In his [2016 GDC talk](#), MTG head designer Mark Rosewater's lesson 18 was that "restrictions breed creativity". Over the history of MTG, players have shown that *cost* restrictions breed *affordable* creativity. For example, Cube designers seek to select sets of 360 or more unique MTG cards that allow interesting gameplay for players that repeatedly draft from the Cube.

Whereas deck construction could be said to invite players to become designers of the games they play, play format design could be said to invite players to become metagame designers. When AI systems of the future are capable of such metametalevel decisions in this imperfect information game of chance, we will see an unprecedented leap forward in game design itself.

In our modest present time, we hope that you and your students find interesting simple research questions to explore in the MTGJSON dataset and sow seeds of curiosity for research advances to come.

References

- Churchill, A., Biderman, S., & Herrick, A. (2019). Magic: The gathering is turing complete. *CoRR*, *abs/1904.09828*. Retrieved from <http://arxiv.org/abs/1904.09828>
- Cowling, P., Ward, C., & Powley, E. (2012). Ensemble determinization in Monte Carlo tree search for the imperfect information card game "Magic: The Gathering". *IEEE Transactions on Computational Intelligence and AI in Games*,

4(4), 241–257. doi: 10.1109/TCIAIG.2012.2204883

- Summerville, A., & Mateas, M. (2016). Mystical tutor: A Magic: The Gathering design assistant via denoising sequence-to-sequence learning. In *Proc. AAAI conference on artificial intelligence and interactive digital entertainment 2016*.
- Ward, C., & Cowling, P. (2009). Monte Carlo search applied to card selection in Magic: The Gathering. In *Proc. 2009 IEEE symposium on computational intelligence and games* (pp. 9–16). IEEE. doi: 10.1109/CIG.2009.5286501



Todd W. Neller is a Professor of Computer Science at Gettysburg College. A game enthusiast, Neller researches game AI techniques and their uses in undergraduate education.