Tab 1

**Predicting the Winner of a VGC Pokémon Battle With Machine Learning**

Jackson Montuoro | Sean Fontaine

**Introduction**

The Video Game Championships, or VGC, is the official format of Nintendo’s Pokémon tournament battles. Our goal for the project is to create a machine learning model which can predict the winner of a battle based on information available in an open team sheet. In this proposal, our goal is to familiarize the audience with competitive pokemon, explain the technical details of our dataset, and enumerate challenges we expect to encounter.

**Overview of Competitive Pokémon**

Pokémon is currently one of the most [recognizable](https://en.wikipedia.org/wiki/List_of_highest-grossing_media_franchises) and popular media franchises. From games, to merchandise, to even [wrapping airplanes](https://en.wikipedia.org/wiki/Pok%C3%A9mon_Jet), Pokémon has maintained its cultural relevance nearly 30 years after the first games. Pokémon Red and Blue were released in 1996, followed soon by the immensely popular Anime and Trading Card Game. Due to the combination of mass appeal, clever marketing, and accessibility to the Pokémon IP, fans new and old continue to enjoy “Pokémania” in increasingly diverse ways.

Pokémon has always had a dedicated *competitive*fan-base. The first official competition, [Nintendo Cup ‘97](https://www.smogon.com/forums/threads/nintendo-cup-97-hub.3682412/), was held locally in Japan as a means of promoting their new games. Even then, Nintendo recognized their games were imbalanced for competitive play, and thus introduced tournament rule restrictions. These rules continued to evolve over the years until crystallizing into the Video Game Championships (VGC) in 2009, the format style we are familiar with to this day.

**What is VGC?**

We now know that VGC is Nintendo’s official competitive format. With [hefty cash prizes](https://www.pokemon.com/us/play-pokemon/pokemon-events/championship-series/2025/world-championships), recognition among the community, and the personal fulfillment of being the “very best,” players are highly motivated to strategize between matches. VGC is fought in a Bring 6, Pick 4 Double Battle format. VGC importantly operates in “regulations,” which changes the pool of available Pokémon every 3-4 months. For our project, we will be focusing on the current regulation, [Regulation I](https://scarletviolet.pokemon.com/en-us/events/regulation-i/), which we will discuss further in subsequent paragraphs.

Beyond this, VGC competitors get the chance to analyze their opponents’ team composition via “open team sheets,” and choose the 4 they theorize will match up well against the other player’s chosen 4. However, how do high ranked players confidently choose which 6 Pokémon to bring before they have even seen their opponent’s composition? What can our machine learning project bring to the table in this aspect?

**What makes a good team?**

To effectively strategize, players must have deep knowledge of battle mechanics, individual Pokémon’s matchups, and how that all fits together in the ever-shifting metagame. In reference to Sun Tzu’s famous quote, the community often says “the battle is won in the team-building phase.” The team building phase differs between competitive formats, but the general approach stays the same. Players combine experience with theory-crafting in order to bring what they consider the strongest subset of available Pokémon to help them win.

**What is the metagame?**

As in any competitive game, community members begin to acknowledge the current “metagame” and adapt their strategies to best take advantage of it. In video games, metagames shift quickly, primarily due to balance updates from developers (or regulation changes in this case). Even chess has changed its official rules throughout the centuries. However, metagames often shift *without* top-down changes. In chess, if a strategy starts to become too dominant, counter-strategies will begin to evolve within the community, which eventually become the new meta staple. The same holds true for competitive Pokémon.

**Related Work**

The [Pokémon Battle Predictor](https://www.pokemonbattlepredictor.com/home) project [began](https://www.reddit.com/r/stunfisk/comments/h9t99t/pokemon_battle_predictor_a_machine_learning/?utm_source=share&utm_medium=web3x&utm_name=web3xcss&utm_term=1&utm_content=share_button) quite similarly to ours, with the intent of predicting the winner of a Pokémon battle. That project evolved into creating an agent with impressive results against human players as well as a browser extension for players. Our project is different in that it focuses on the 2 vs 2 double battle format. It is also different in that predictions will be made before a single turn has been played, and not from any choices or events that occur mid-battle.

[AlphaGO](https://deepmind.google/research/projects/alphago/) was developed by Google Deepmind to beat top human players in Go, a game with immense strategic depth and branching complexity. AlphaGO shows that complex, partially observable environments with incredibly variable action spaces can still be modeled effectively with Machine Learning. One difference is that upon board initialization, the pieces of Go begin neutrally, whereas individual Pokémon may be more or less valuable depending on the matchup. Where AlphaGO observes every individual game state, our project is largely dependent on team composition, or “board initialization.”

One last project similar to ours is [Catan Analytics by Duddhawork](https://duddhawork.com/blog/catan-analytics-how-to-win-with-data-driven-strategies/), which examined 50 four-player Catan games to identify factors influencing win rates. Key findings included the impact of settlement placement order, resource abundance, and resource diversity on game outcomes. Like our project, this one is focused on identifying winning strategies just based on the initialization of a stochastic game. Where we plan to improve on this project is instead of relying on exploratory data analysis to drive our claims, we will build a model in order to predict victory.

**Data Source**

Our dataset consists of 9,773 parsed battle logs accessed from Pokémon Showdown’s GitHub, the leading online simulator for competitive Pokémon formats including VGC. Each log captures complete team compositions for both players, including species, items, abilities, and four moves per Pokémon. The dataset also records the selected leads for each player and the outcome (p1\_win). This dataset is cleanly structured and is well suited for both supervised classification tasks and unsupervised exploration of team archetypes. All player usernames have been retained only as anonymized string identifiers, and battles pertain to Regulation I of VGC.

| **Column Group** | **Description** | **Example** |
| --- | --- | --- |
| p1\_species\_\* | Species name of each Pokémon on P1’s team | Incineroar |
| p1\_item\_\* | Held item for each Pokémon on P1’s team | Leftovers |
| p1\_move\_\*\_\* | Moves 1-4 for each Pokémon on P1’s team | Fake Out |
| p1\_ability\_\* | Ability of each Pokémon on P1’s team | Intimidate |
| p1\_lead\_1/2 | The Pokémon P1 selected to lead | Miraidon, Iron Hands |
| p1\_win | Whether P1 won the battle | 0 or 1 |
| (same for P2) | Identical structure for P2 |  |

**Feature Engineering**

To prepare for both supervised and unsupervised machine learning tasks, we will begin by extracting structured features from the raw battle logs. Each record contains full team compositions for both players, including species, items, abilities, and four moves per Pokémon. From this, we are generating a structured feature matrix suitable for model input.

**Preprocessing**

The raw data from Pokémon Showdown is well-formatted and is expected to require minimal cleaning. We plan to search for missing values in columns such as species or result (p1\_win). Player usernames are preserved only as anonymized string and are not included in the feature set- this would likely cause data leakage.

**Final Features**

We are engineering multiple types of features to capture both individual Pokémon attributes and team-level dynamics:

* **Categorical Features (One-Hot Encoded)**
  + Pokémon species (e.g., Flutter Mane, Gothitelle) for both players
  + Items held by each Pokémon
  + Abilities of each Pokémon
  + Selected leads (p1\_lead\_1, p1\_lead\_2, etc.)
* **Derived Features (Planned)**
  + **Type coverage vector:** a multi-hot encoded vector indicating the types represented on a team (Electric, Dragon)
  + **Team synergy score:** indicators based on known strong pairings in the metagame (Electric Seed Farigiraf + Miraidon).
  + **Speed control presence:** binary feature indicating presence of common moves like Trick Room, Tailwind, or Icy Wind
  + **Support score:** binary flags for the presence of Pokémon capable of setting terrain/weather or redirection moves like Follow Me.
* **Positional Features:**
  + Team Slot indicators (e.g. is Flutter Mane leading?)
  + Lead Pokémon overlap between teams (e.g. both teams lead with Amoonguss)
* **Match-Level Features:**
  + Shared Pokémon count between teams (0-6)
  + Shared items on shared Pokémon between teams (0-6)
  + Shared abilities on shared Pokémon between teams
  + Usage rate (how often do we see this species in the dataset?)

**Challenges**

Pokémon VGC is a complex game to model. During team building, players choose a team from over 1000 Pokémon, each with unique stats, moves, abilities, and types. Pokémon have 4 different moves to use per turn, and the same Pokémon can perform wildly differently between matches depending on training (EV distribution), items, movesets, and synergy with other Pokémon. VGC is played in a bring-6 pick-4 doubles format, expanding the total options available per turn and before-battle multiplicatively. As for other challenges, Pokémon battles include random factors like critical hits, move accuracy, and secondary effects, which complicates modeling. This introduces a stochastic element and makes it so no battle is ever deterministic. Finally, rule changes called ‘regulations’ occur every 3 to 4 months, massively influencing the popularity and effectiveness of different Pokémon and strategies.

**Part A (Supervised Learning)**

**Methods Description**

**Supervised Evaluation**

**Failure Analysis**

Data consists of 5000+ battle logs scraped from Pokémon Showdown, a leading online Pokémon battle hosting site. We plan to run logistic regression, random forests, and eventually neural networks, to predict the outcomes in a probabilistic range. We aim to output win probabilities over binary outcomes, as Pokémon is a stochastic game and even the best players get unlucky or have bad games. As for evaluation, we are going to focus on metrics such as ROC-AUC score and F1-Score. For our final visualizations, I would like to take aesthetics into consideration, representing Pokémon by their official artwork via [PokeAPI](https://pokeapi.co/) calls, similar to [Poképaste](https://pokepast.es/6d4003745e66f0ed). I imagine both teams on each side represented by their sprites, with a % figure indicating the likelihood of winning for both teams.

**Part B (Unsupervised Learning)**

**Methods Description**

**Unsupervised Evaluation**

Dataset is the same as in the supervised portion. We want to use unsupervised learning to identify common team compositions so that we can a) check the extent to which these clusters align with ladder win rates or tournament success b) add team composition variables as features to improve our supervised predictions. We will need to one-hot encode categorical features such as Pokémon species, items, and abilities. We will need to remove trainer identifiers as well as outcome labels of battles. Dimensionality reduction may be necessary for visualization and performance. We plan to use K-means clustering to identify team composition. We plan to use PCA for dimensionality reduction. These techniques will be helpful for interpreting high-dimensional data like Pokémon teams. Similar to Part A, we will likely access [PokeAPI](https://pokeapi.co/) for visualization purposes to highlight prominent pokémon within archetype clusters. To evaluate these clusters, we will manually cross reference them against existing, known archetypes and synergies. To make visualizations more attractive, we will consider t-SNE.

**Discussion**

**Ethical Considerations**

**Statement of Work**

**References**

**Project Submission**

**Team Planning:**

By the end of **week 3**, the two of us will have collaborated on similar parts of exploratory data analysis. By the end of **week 4**, Sean will have experimented with model parameters in order to gain insights via supervised learning. Jackson will have discovered insights into common clusters. By the end of **Week 6,** our model will be at least 95% ready for production. We will focus the last two weeks on final model touchess and cleaning up visualizations. We would like to dedicate the last week to presentation creation.