Pokemon Machine Learning

Final Presentation

Brian Gerrard | Linkedin | Github
Jennifer Cheng | LinkedIn | Github
Tatiana Kalainoff | LinkedIn | Github
Project Repository

Agenda

- Meet the Team
- About the Project
  - Mission
  - About the Data
  - Dashboard Outline
- A Closer Look
  - Evolution of Pokemon
  - PokeBadges
  - Build A Pokemon
- Further Exploration

Meet Team Sprocket

Brian Gerrard

Linkedin | Github

Jennifer Cheng

LinkedIn | Github

Tatiana Kalainoff

LinkedIn | Github

We are students of the
Columbia University | Engineering Data Analytics and Visualization Certification Program
Class of August 2021

We’re showcasing a data-led retrospective on 7 Generations of Pokemon

Happy Birthday, Pokemon!

Why we love this project

Pokemon has had a cultural resurgence in recent years, all leading to the 25th Anniversary of Pokemon in October 2021.

This franchise is a cultural phenomenon, and, in 2021, it is celebrating a milestone anniversary while consumers are yearning for nostalgia and simple pleasures.

We're taking a look at how the original Pokedex has evolved across 7 Generations. Pun intended.

The Data

We're using a robust dataset that contains a catalogue of Pokemon that spans the first 7 Generations.

Some stats we're able to analyze are outlined here ➜

Data Column	Description
Pokedex Data	A catalogue of all detailed Pokemon species and features
Base Stats Data	Each individual value that determines a Pokemon's skills, strengths, weaknessess and overall abilities
Training Data	Quantified efforts for leveling up, upskilling and evolving your pokemon
Breeding Data	Detailed data on which types of Pokemon species can breed and what they evolve into
Type Defenses Data	Data that determines a Pokemon's resistance against physical attacks

Learn more about Pokemon stats

Storytelling with Data

We built three different data-led modules for a fun user experience

Web-Based Dashboard

Skills:

JavaScript, HTML, Python, Flask, D3

The Evolution of Pokemon

A Retrospective by the Numbers

Skills:

D3, JavaScript, HTML, Tableau, SQL

Build A Pokemon

An Interactive AI Experience

Skills:

Postgres, SQL, SQLAlchemy, Python, Data Cleaning, Supervised Machine Learning (data pre-processing, algorithm testing)

Evolution of Pokemon

A Retrospective by the Numbers

```
// created circle for every datapoint
var circles = svg.selectAll(".poke_type")
    .data(datapoints)
    .enter().append("circle")
    .attr("class", "poke_type")
    /*.attr("r", 10)*/
    .attr("r", function(d) {
        return radiusScale(d.hp)
    })
    .attr("fill", function(d) {
        return typesToColors[d.type_1]
    })
    .on('click', function(d) {
        console.log(d)
    })

d3.select("#friendly").on('click', function() {
    simulation
    .force("x", forceXfriendly)
    .alphaTarget(0.2)
    .restart()
})

d3.select('#return').on('click', function() {
    simulation
        .force("x", forceXreturn)
        .alphaTarget(0.01)
        .restart()
})

// for each datapoint, look back at the forces
simulation.nodes(datapoints)
    .on('tick', ticked)

// and reposition circle
function ticked() {
    circles
        .attr("cx", function(d) {
```

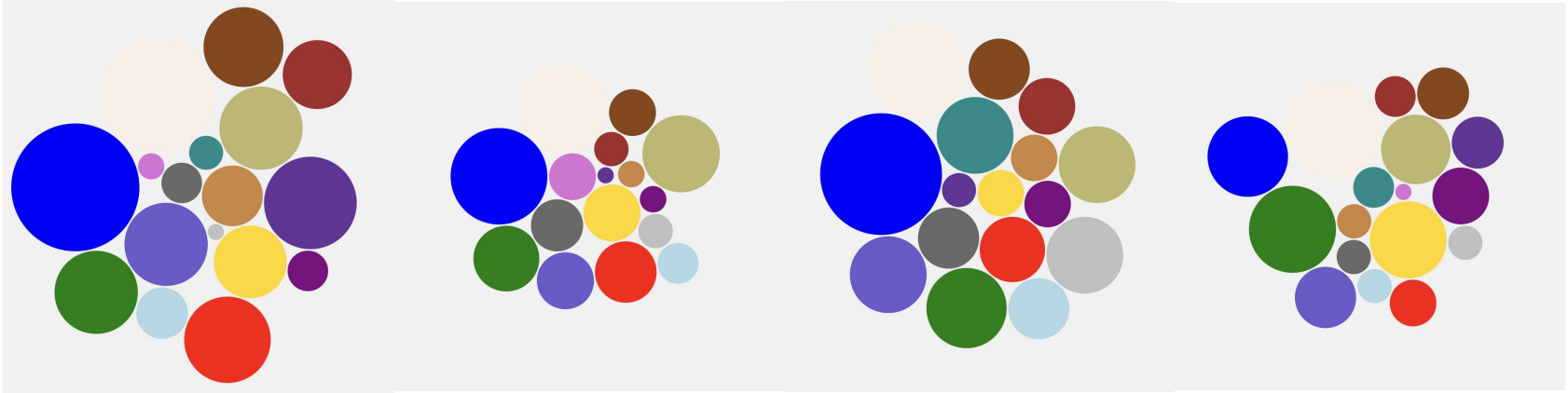
Inspiration:https://www.bloomberg.com/graphics/2015-auto-sales/

Data Preparation: Addressing null values

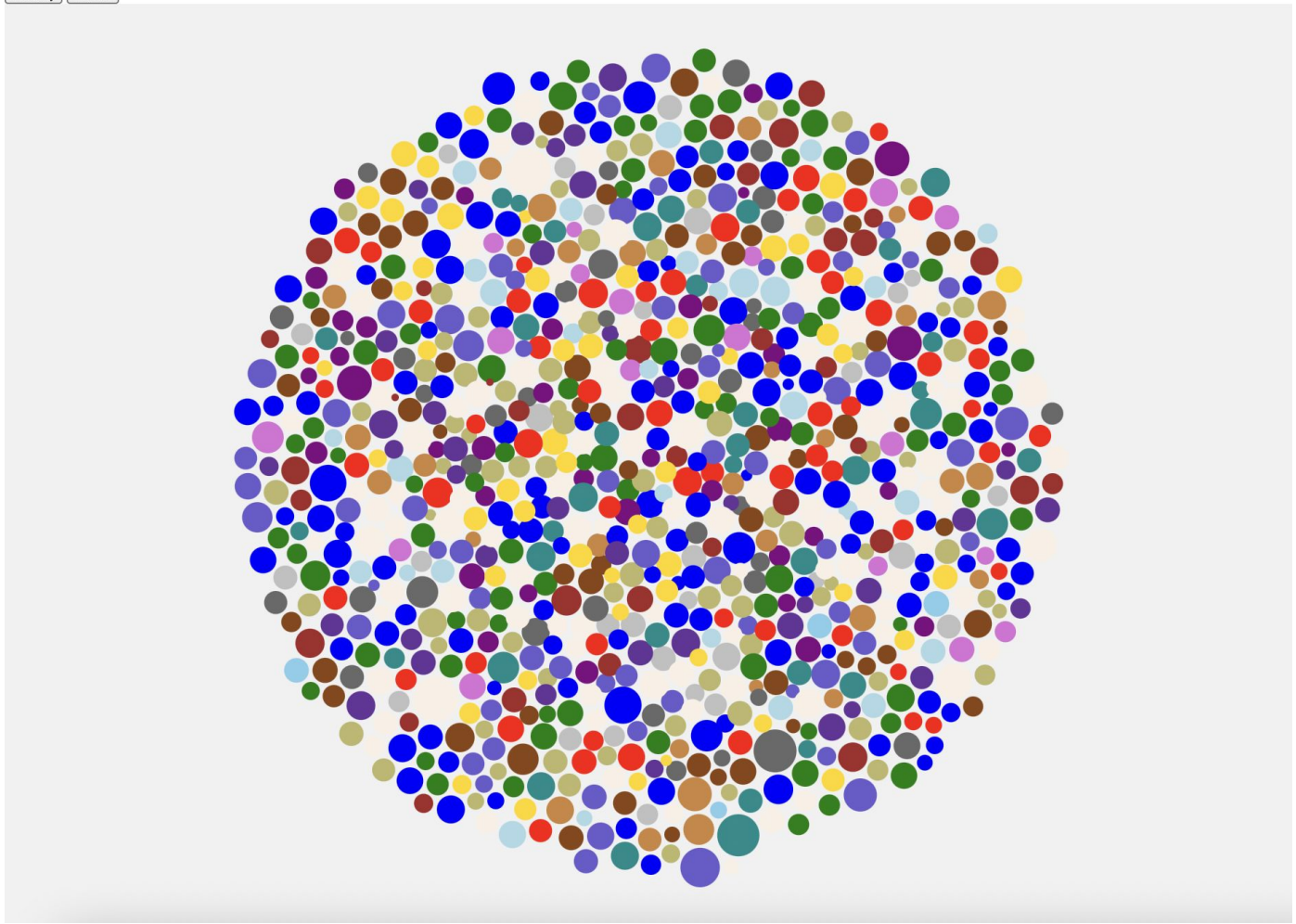
Data Exploration: Bringing in additional columns for analysis

Coding Languages used:

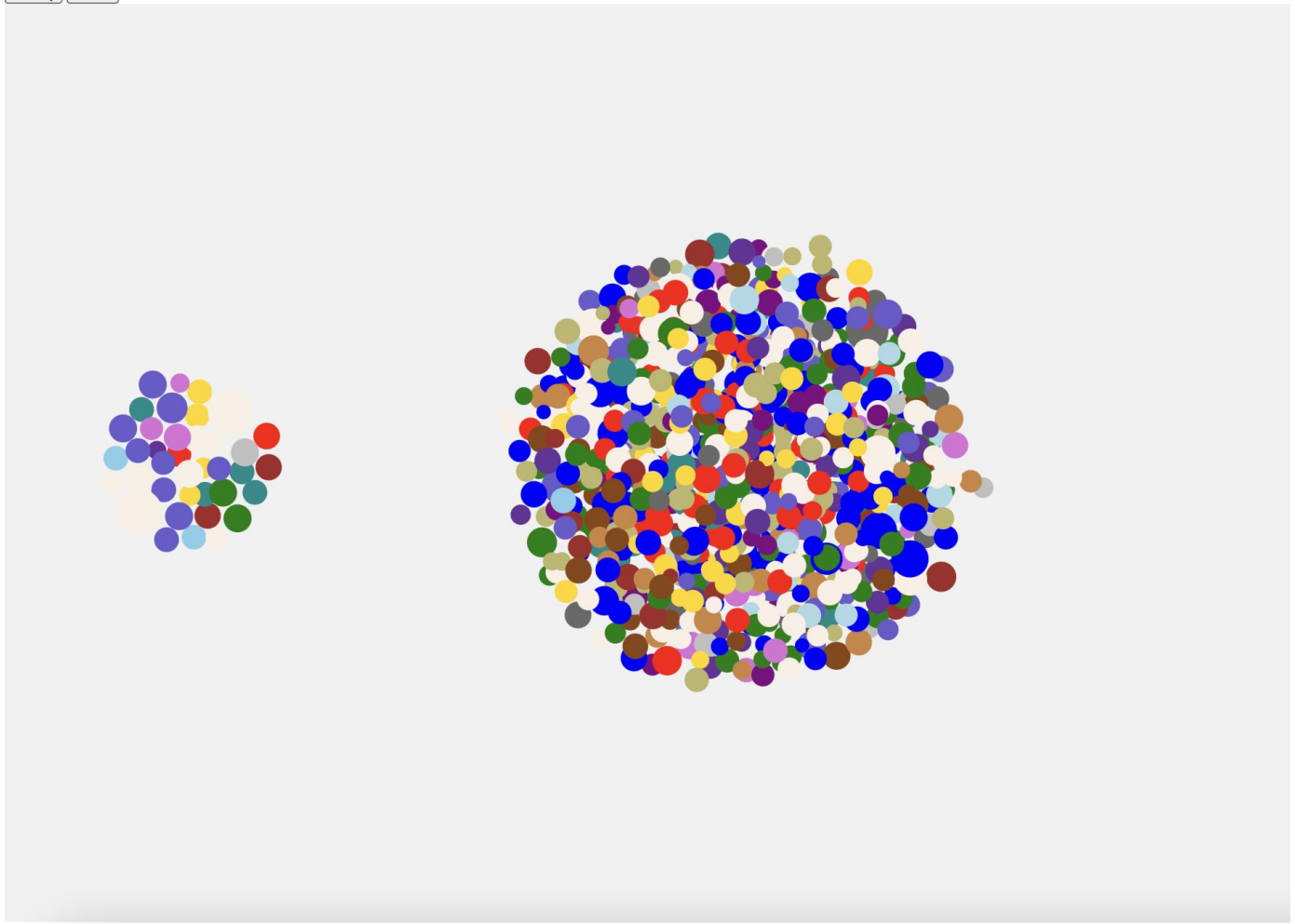
- A - SQL to explore data
- B - HTML, Javascript, D3 for specific interactive visualizations
- C - Tableau for larger scale data trends



Friendly Return

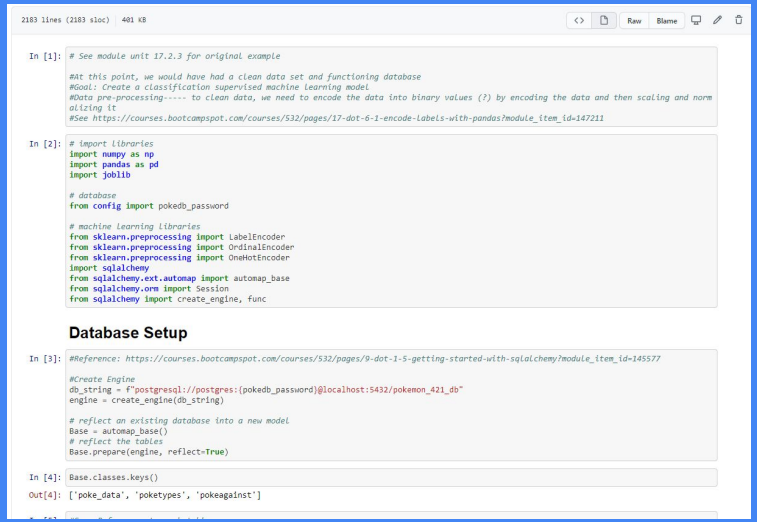


Friendly Return



Build a Pokemon

[Part I: Machine Learning]



2183 lines (2183 sloc) | 401 KB

Raw | Blame

In [1]:
```
# See module unit 17.2.3 for original example

#At this point, we would have had a clean data set and functioning database
#Goal: Create a classification supervised machine learning model
#Data pre-processing----- to clean data, we need to encode the data into binary values (?) by encoding the data and then scaling and normalizing it
#See https://courses.bootcampspot.com/courses/532/pages/17-dot-6-1-encode-labels-with-pandas?module_item_id=147211
```

In [2]:
```
# import libraries
import numpy as np
import pandas as pd
import joblib

# database
from config import pokedb_password

# machine learning libraries
from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import OrdinalEncoder
from sklearn.preprocessing import OneHotEncoder
import sqlalchemy
from sqlalchemy.ext.automap import automap_base
from sqlalchemy.orm import Session
from sqlalchemy import create_engine, func
```

Database Setup

In [3]:
```
#Reference: https://courses.bootcampspot.com/courses/532/pages/9-dot-1-5-getting-started-with-sqlalchemy?module_item_id=145577

#Create Engine
db_string = f"postgresql://postgres:{pokedb_password}@localhost:5432/pokemon_421_db"
engine = create_engine(db_string)

# reflect an existing database into a new model
Base = automap_base()
# reflect the tables
Base.prepare(engine, reflect=True)
```

In [4]:
```
Base.classes.keys()
```

Out[4]:
```
['poke_data', 'poketypes', 'pokeagainst']
```

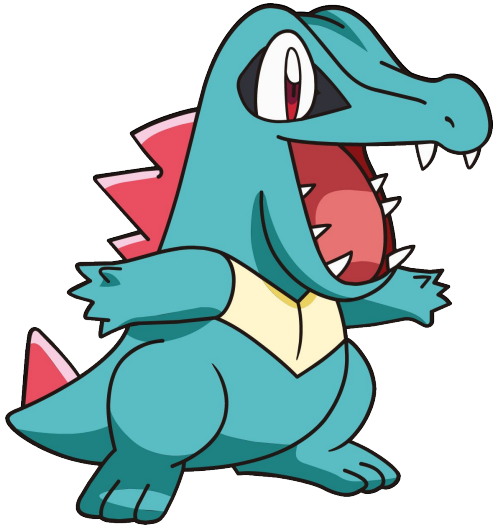
- Data Preparation:
  - Cleaning data
    - Addressing null values
    - Create uniform data types
    - Encoding data
    - Data pre-processing for Machine Learning
  - Training Machine Learning Model
    - Python, SKLearn, SQLAlchemy

Build a Pokemon

Part II: Building a User Experience

- Using Flask and HTML to build user-input fields
- Ingest and organize data into processable form to pass through machine learning module

What's Next?



More robust animation using Plotly or D3

Web scrape for images of similar Pokemon upon user input submission

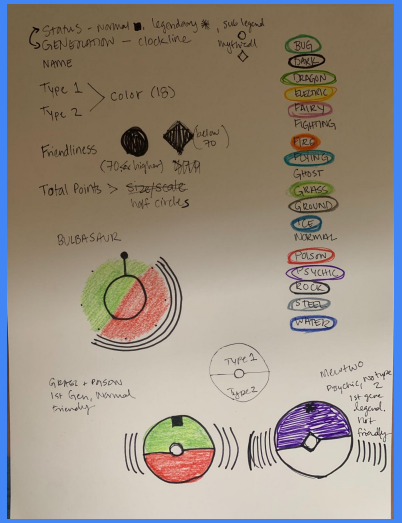
---

Exploring more machine learning algorithms

Suggestions?

PokeBadges

Stats at a Glance



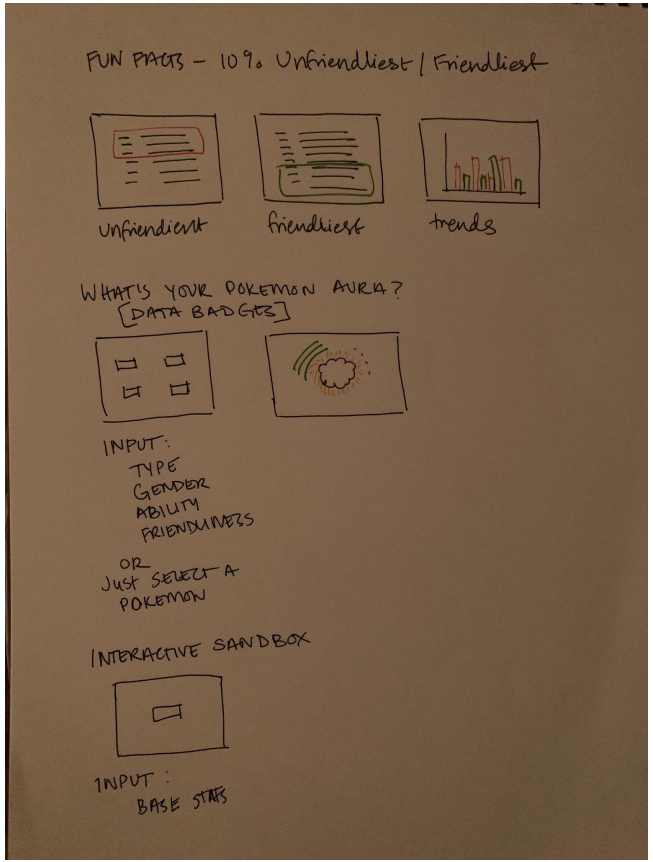
Inspiration: https://shirleywu.studio/filmflowers/

Data Preparation: Addressing null values

Data Exploration: Visualize stats of an individual pokemon at a glance

Coding Languages used: TBD

Appendix



Storyboarding

[Click here to see the full ERD on the Github]

Database ERD