PGA GOLF SIMULATOR

Shusuke Hashimoto

L435/L555

CONTENTS

01 METHODS

Datasets (CSV)
Python libraries
Statistical models (Distributions)
Binary decision modeling

02 RESULTS

Fine-tuning distributions
Visualizations

03 ISSUES

Putting model limitation
Playoff mechanism

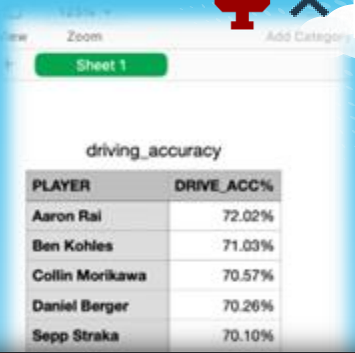
04 DISCUSSION

Machine learning applications

01 METHODS

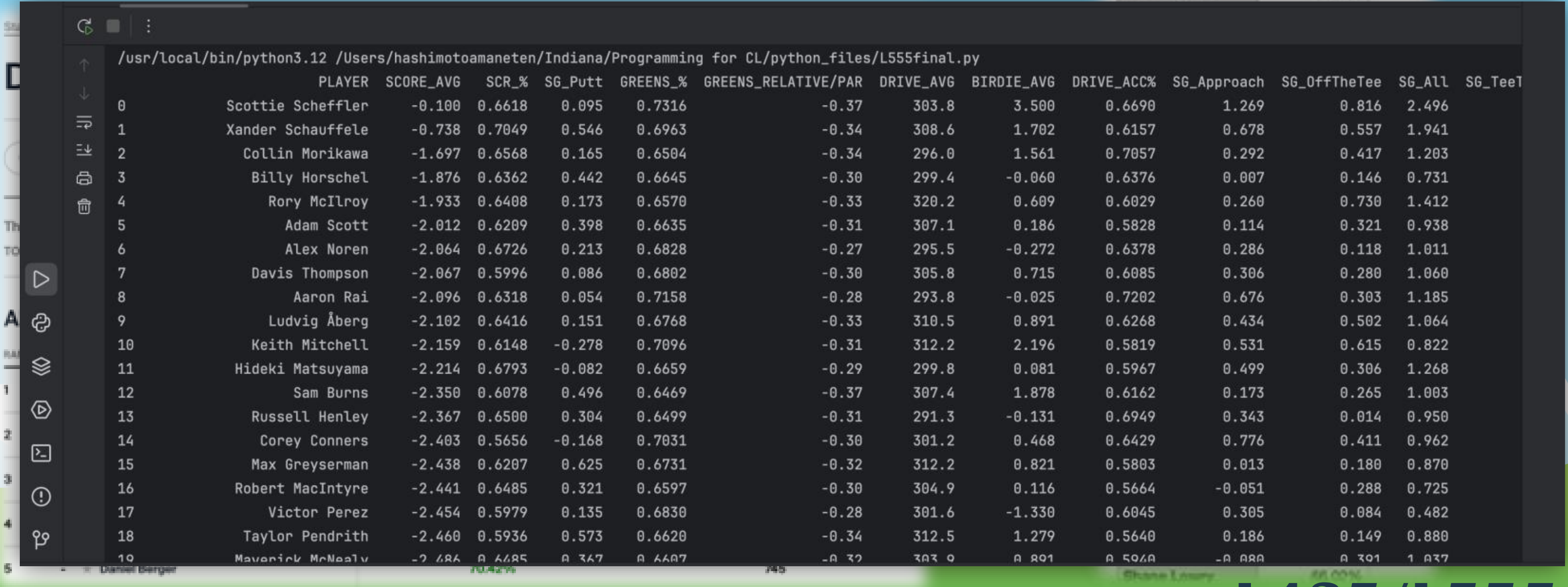
Statistical datasets

extracted from the PGA official webpage



driving_accuracy

PLAYER	DRIVE_ACC%
Aaron Rai	72.02%
Ben Kohles	71.03%
Collin Morikawa	70.57%
Daniel Berger	70.26%
Sepp Straka	70.10%



```
/usr/local/bin/python3.12 /Users/hashimotoamaneten/Indiana/Programming for CL/python_files/L555final.py
                PLAYER  SCORE_AVG   SCR_%  SG_Putt  GREENS_%  GREENS_RELATIVE/PAR  DRIVE_AVG  BIRDIE_AVG  DRIVE_ACC%  SG_Approach  SG_OffTheTee  SG_All  SG_TeeT
0     Scottie Scheffler     -0.100  0.6618    0.095    0.7316                -0.37      303.8       3.500      0.6690        1.269         0.816   2.496
1      Xander Schauffele     -0.738  0.7049    0.546    0.6963                -0.34      308.6       1.702      0.6157        0.678         0.557   1.941
2        Collin Morikawa     -1.697  0.6568    0.165    0.6504                -0.34      296.0       1.561      0.7057        0.292         0.417   1.203
3         Billy Horschel     -1.876  0.6362    0.442    0.6645                -0.30      299.4      -0.060      0.6376        0.007         0.146   0.731
4           Rory McIlroy     -1.933  0.6408    0.173    0.6570                -0.33      320.2       0.609      0.6029        0.260         0.730   1.412
5             Adam Scott     -2.012  0.6209    0.398    0.6635                -0.31      307.1       0.186      0.5828        0.114         0.321   0.938
6             Alex Noren     -2.064  0.6726    0.213    0.6828                -0.27      295.5      -0.272      0.6378        0.286         0.118   1.011
7         Davis Thompson     -2.067  0.5996    0.086    0.6802                -0.30      305.8       0.715      0.6085        0.306         0.280   1.060
8              Aaron Rai     -2.096  0.6318    0.054    0.7158                -0.28      293.8      -0.025      0.7202        0.676         0.303   1.185
9           Ludvig Åberg     -2.102  0.6416    0.151    0.6768                -0.33      310.5       0.891      0.6268        0.434         0.502   1.064
10        Keith Mitchell     -2.159  0.6148   -0.278    0.7096                -0.31      312.2       2.196      0.5819        0.531         0.615   0.822
11      Hideki Matsuyama     -2.214  0.6793   -0.082    0.6659                -0.29      299.8       0.081      0.5967        0.499         0.306   1.268
12             Sam Burns     -2.350  0.6078    0.496    0.6469                -0.37      307.4       1.878      0.6162        0.173         0.265   1.003
13        Russell Henley     -2.367  0.6500    0.304    0.6499                -0.31      291.3      -0.131      0.6949        0.343         0.014   0.950
14         Corey Conners     -2.403  0.5656   -0.168    0.7031                -0.30      301.2       0.468      0.6429        0.776         0.411   0.962
15        Max Greyserman     -2.438  0.6207    0.625    0.6731                -0.32      312.2       0.821      0.5803        0.013         0.180   0.870
16      Robert MacIntyre     -2.441  0.6485    0.321    0.6597                -0.30      304.9       0.116      0.5664       -0.051         0.288   0.725
17          Victor Perez     -2.454  0.5979    0.135    0.6830                -0.28      301.6      -1.330      0.6045        0.305         0.084   0.482
18       Taylor Pendrith     -2.460  0.5936    0.573    0.6620                -0.34      312.5       1.279      0.5640        0.186         0.149   0.880
19      Maverick McNealy     -2.486  0.6485    0.367    0.6607                -0.32      303.9       0.891      0.5940       -0.080         0.391   1.037
```

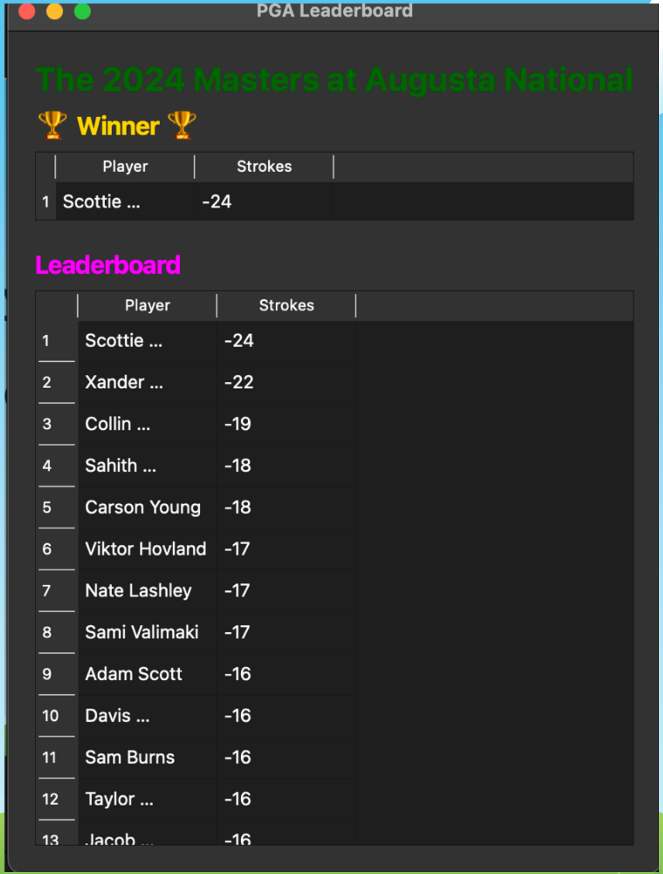
01 METHODS

Libraries

NumPy, pandas (Manipulating/Analyzing data)

random (For stochastic purpose)

PyQt5.QtWidgets (Leaderboard visualization)



PGA Leaderboard

The 2024 Masters at Augusta National

🏆 Winner 🏆

	Player	Strokes
1	Scottie ...	-24

Leaderboard

	Player	Strokes
1	Scottie ...	-24
2	Xander ...	-22
3	Collin ...	-19
4	Sahith ...	-18
5	Carson Young	-18
6	Viktor Hovland	-17
7	Nate Lashley	-17
8	Sami Valimaki	-17
9	Adam Scott	-16
10	Davis ...	-16
11	Sam Burns	-16
12	Taylor ...	-16
13	Jacob ...	-16

L435/L555

01 METHODS

Distribution types

- Lognormal distribution
- Truncated normal distribution
- Uniform distribution

What are they?

Which type is favorable for which scenario?
(driver, shot to the green, approach, layup. putt...)

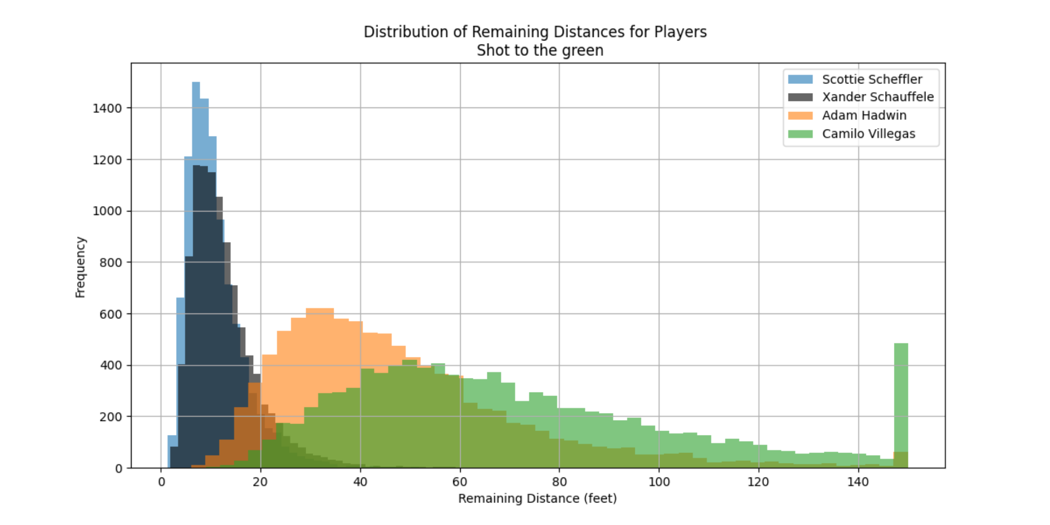
01 METHODS

Lognormal Distribution

The data where the logarithm of the variable follows a normal distribution

- ✓ right-skewed
- ✓ cannot have negative values
  => preferable for distance

Shot to the green

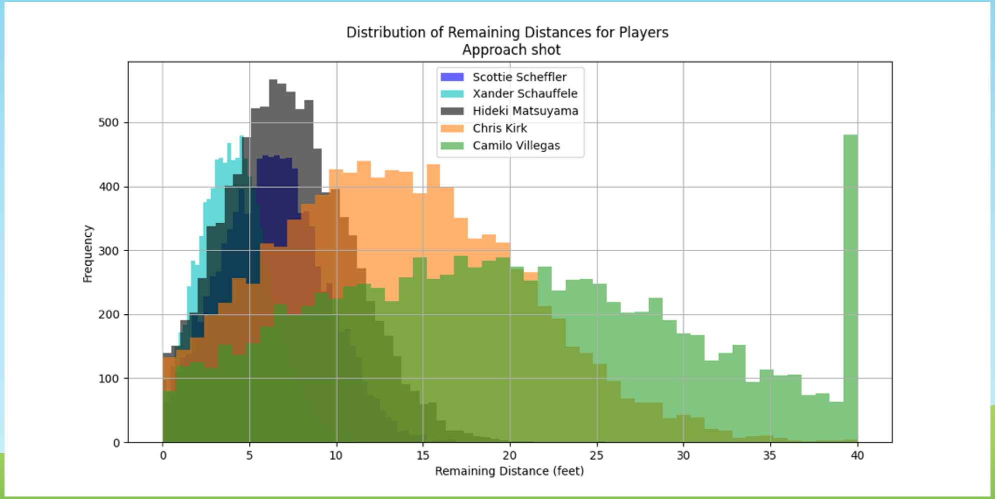


01 METHODS

Truncated Normal Distribution

"cut off" (truncated) to fit within a specified range.

- ✓ values stay within realistic limits
- ✓ avoid being unrealistically far away from the hole



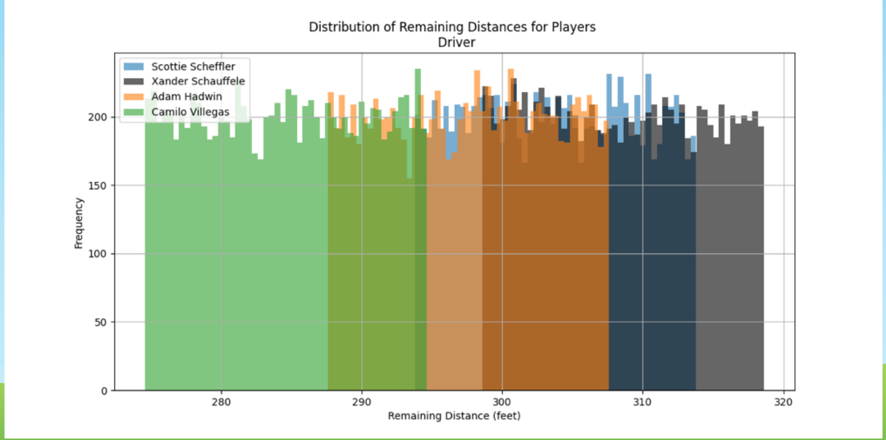
01 METHODS

Uniform Distribution

equal probability to all values within a specified range

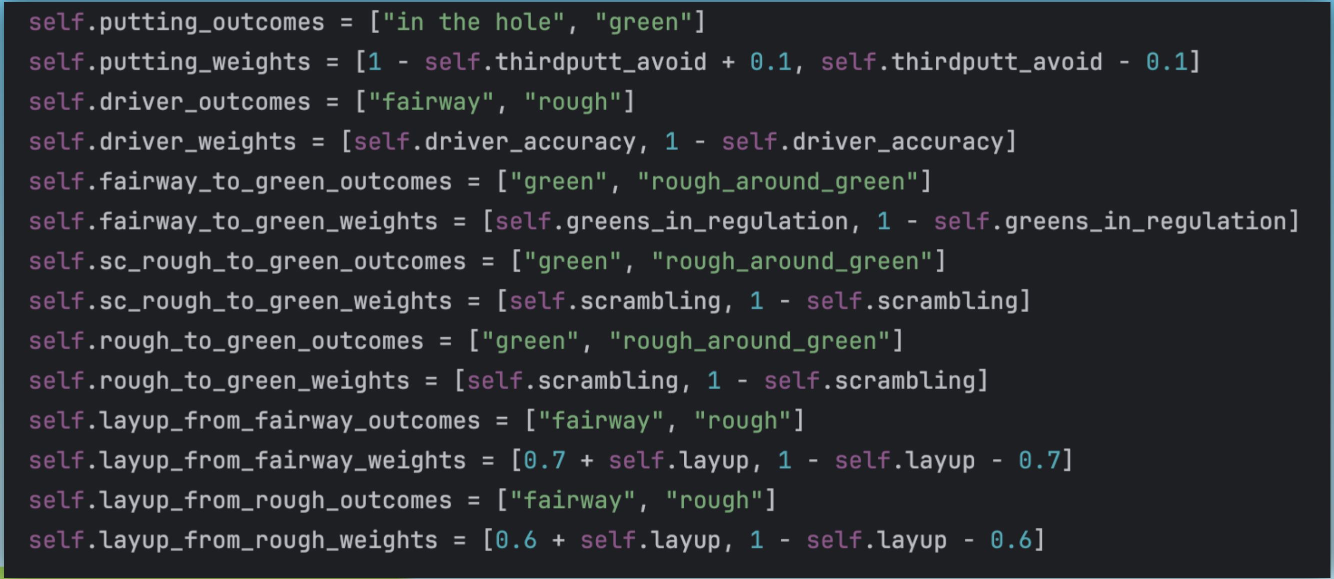
- ✓ Every outcome within the range is equally likely
- ✓ Highly consistent

Driver



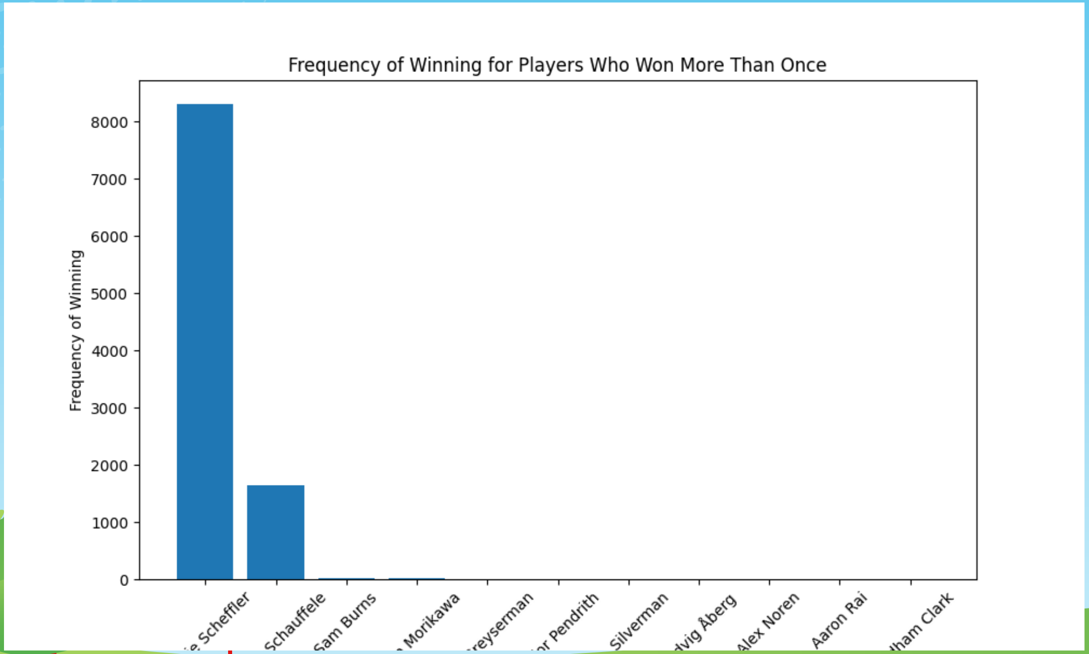
01 METHODS

Binary Decision Modeling



```
self.putting_outcomes = ["in the hole", "green"]
self.putting_weights = [1 - self.thirdputt_avoid + 0.1, self.thirdputt_avoid - 0.1]
self.driver_outcomes = ["fairway", "rough"]
self.driver_weights = [self.driver_accuracy, 1 - self.driver_accuracy]
self.fairway_to_green_outcomes = ["green", "rough_around_green"]
self.fairway_to_green_weights = [self.greens_in_regulation, 1 - self.greens_in_regulation]
self.sc_rough_to_green_outcomes = ["green", "rough_around_green"]
self.sc_rough_to_green_weights = [self.scrambling, 1 - self.scrambling]
self.rough_to_green_outcomes = ["green", "rough_around_green"]
self.rough_to_green_weights = [self.scrambling, 1 - self.scrambling]
self.layup_from_fairway_outcomes = ["fairway", "rough"]
self.layup_from_fairway_weights = [0.7 + self.layup, 1 - self.layup - 0.7]
self.layup_from_rough_outcomes = ["fairway", "rough"]
self.layup_from_rough_weights = [0.6 + self.layup, 1 - self.layup - 0.6]
```

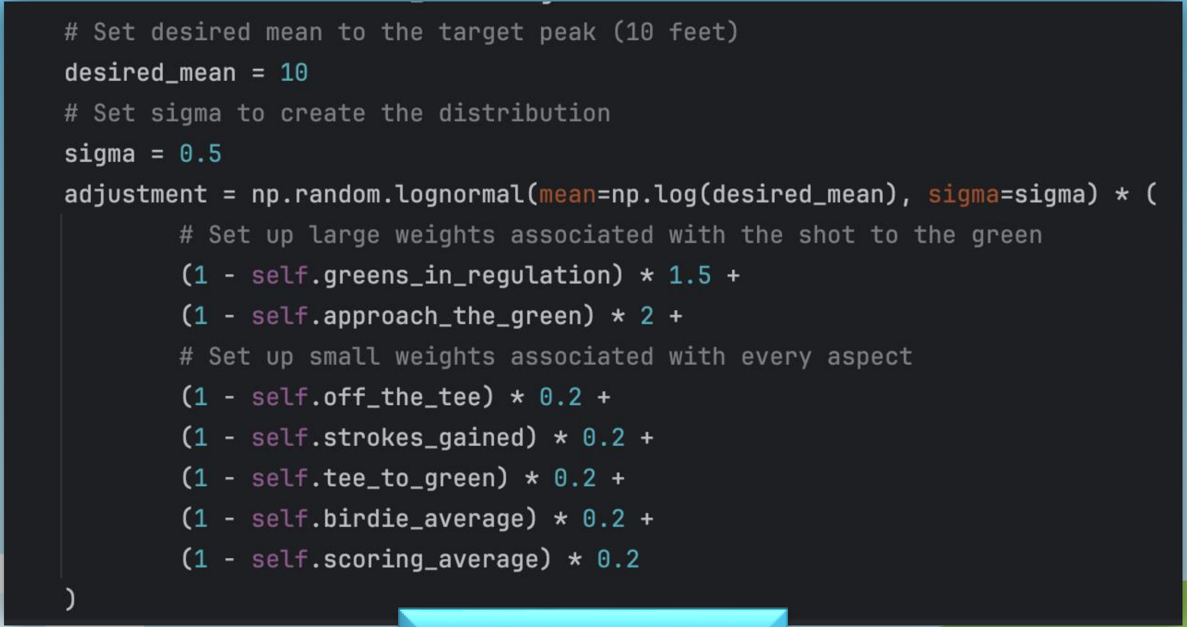
02 RESULTS



PERFECT!

03 ISSUES

Fine-tuning



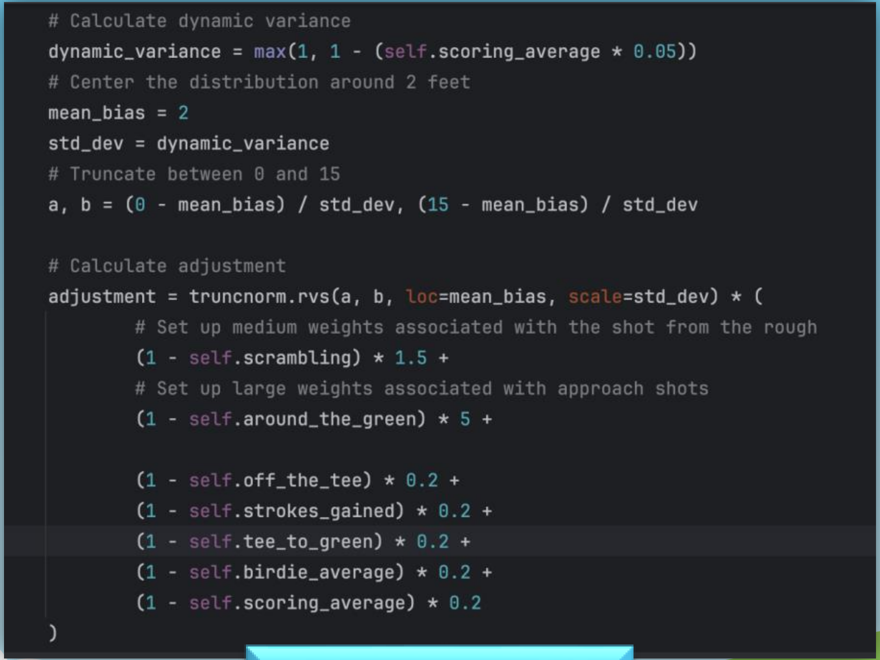
```
# Set desired mean to the target peak (10 feet)
desired_mean = 10
# Set sigma to create the distribution
sigma = 0.5
adjustment = np.random.lognormal(mean=np.log(desired_mean), sigma=sigma) * (
    # Set up large weights associated with the shot to the green
    (1 - self.greens_in_regulation) * 1.5 +
    (1 - self.approach_the_green) * 2 +
    # Set up small weights associated with every aspect
    (1 - self.off_the_tee) * 0.2 +
    (1 - self.strokes_gained) * 0.2 +
    (1 - self.tee_to_green) * 0.2 +
    (1 - self.birdie_average) * 0.2 +
    (1 - self.scoring_average) * 0.2
)
```

Shot to the green

L435/L555

03 ISSUES

Fine-tuning

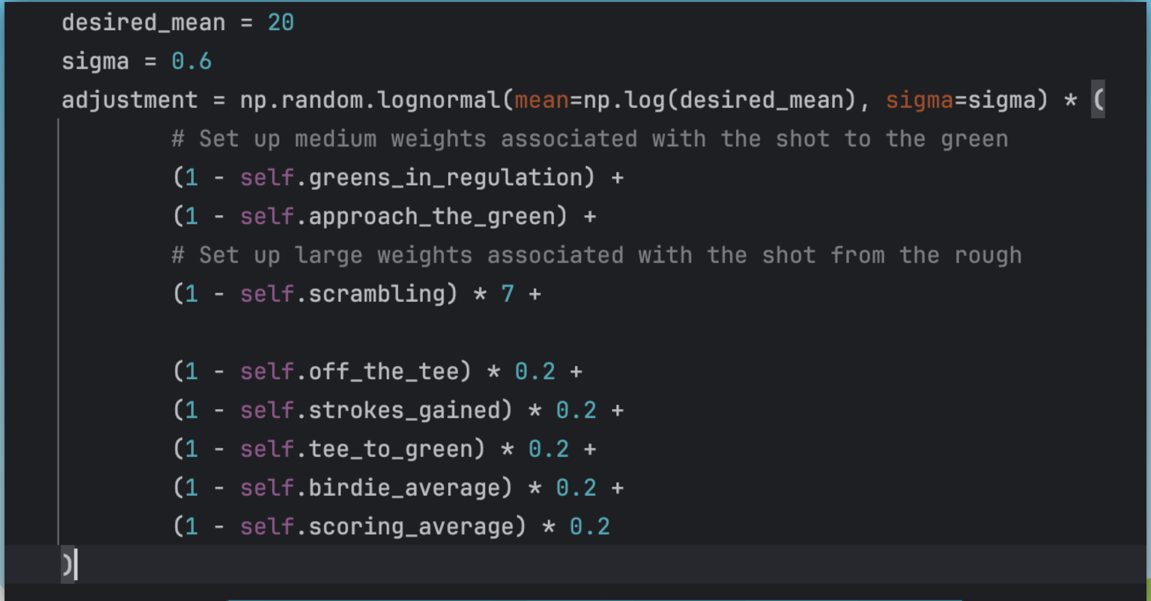


```
# Calculate dynamic variance
dynamic_variance = max(1, 1 - (self.scoring_average * 0.05))
# Center the distribution around 2 feet
mean_bias = 2
std_dev = dynamic_variance
# Truncate between 0 and 15
a, b = (0 - mean_bias) / std_dev, (15 - mean_bias) / std_dev

# Calculate adjustment
adjustment = truncnorm.rvs(a, b, loc=mean_bias, scale=std_dev) * (
        # Set up medium weights associated with the shot from the rough
        (1 - self.scrambling) * 1.5 +
        # Set up large weights associated with approach shots
        (1 - self.around_the_green) * 5 +

        (1 - self.off_the_tee) * 0.2 +
        (1 - self.strokes_gained) * 0.2 +
        (1 - self.tee_to_green) * 0.2 +
        (1 - self.birdie_average) * 0.2 +
        (1 - self.scoring_average) * 0.2
)
```

Approach

03 ISSUES

Fine-tuning

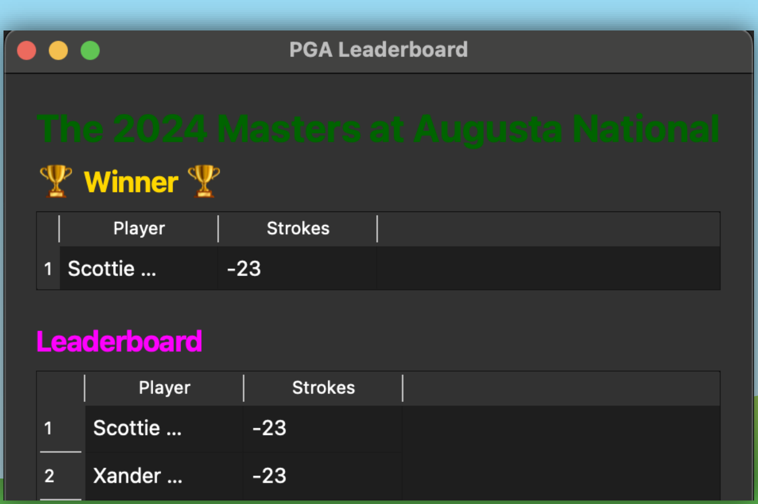


```
desired_mean = 20
sigma = 0.6
adjustment = np.random.lognormal(mean=np.log(desired_mean), sigma=sigma) * (
        # Set up medium weights associated with the shot to the green
        (1 - self.greens_in_regulation) +
        (1 - self.approach_the_green) +
        # Set up large weights associated with the shot from the rough
        (1 - self.scrambling) * 7 +

        (1 - self.off_the_tee) * 0.2 +
        (1 - self.strokes_gained) * 0.2 +
        (1 - self.tee_to_green) * 0.2 +
        (1 - self.birdie_average) * 0.2 +
        (1 - self.scoring_average) * 0.2
)
```

Shot to the green from the rough

04 DISCUSSION

- Play-off scenario
- Machine learning for predictive analytics



The 2024 Masters at Augusta National

🏆 Winner 🏆

	Player	Strokes
1	Scottie ...	-23

Leaderboard

	Player	Strokes
1	Scottie ...	-23
2	Xander ...	-23

THANKS!

DO YOU HAVE ANY QUESTIONS?

L435/L555