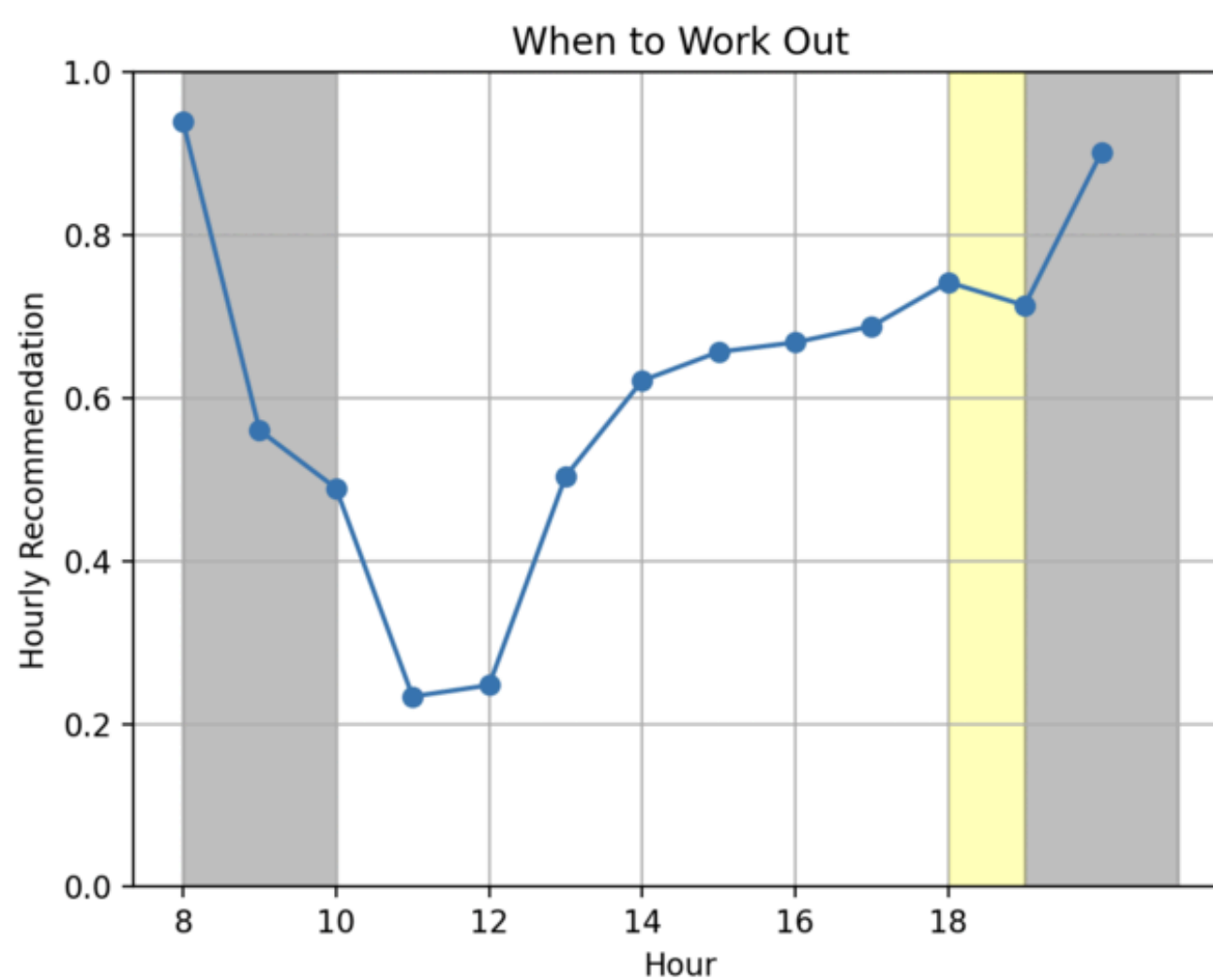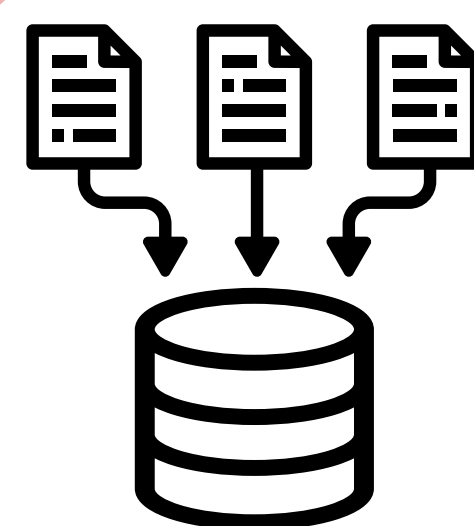# UCSBusy

By providing **hour-based predictions** of crowd sizes, we aim to help you **plan** your workouts better, **avoid long waits**, and have a more **enjoyable experience**.

James Cha
Quinn Koster
Alex Morifusa
Seiichi Nakahira
Christina Hefan Cui

Output

When to Work Out — Hourly Recommendation vs. Hour

1. Problem

- The unpredictability of crowd sizes at different times at the gym was a problem that needed a solution.
- That's when we decided to leverage our knowledge of statistical modeling to predict the least busy times at the Rec Center.
- We created UCSBusy with a single goal in mind - to help you make the most of your time at the Rec Center.

2. Data

- Started by acquiring data directly from the UCSB Recreation Center.
- Data included information about the number of people visiting the center at different times and on different days.
- Exploratory Data Analysis helped us identify trends; notable finding: Sundays have half the number of people compared to Mondays.
- Powered by these discoveries, we wrote a Python script to suggest gym times considering peak hours, least busy days, and the impact of special events or holidays on crowd sizes.

3. Model

- Started with data from the UCSB Rec Center and used Python for analysis.
- Initial approach with a normal distribution didn't fit well due to the time series nature of our data.
- Discovered Poisson models, which were a better fit.
- Data had many zeroes due to early-morning openings/closures and special holidays.
- Incorporated a Zero-Inflated Poisson model to handle excess zeroes and improve prediction accuracy.
- Took the lambda values from the Poisson and ZIP models and stored them in the backend for future calculations.
- This blend of techniques powers UCSBusy's reliable predictions.

4. Calendar Function

- Python script incorporates a powerful calendar function.
- Utilize lambda values stored in the backend derived from Poisson and Zero-Inflated Poisson models.
- Apply lambda values to a Poisson distribution, assuming $k = 0$, to predict crowd size at any given hour.
- Implemented a penalty function considering the user's daily schedule into calculations to suggest optimal gym times.
- Normalize values before combining them to ensure accuracy.

Input

```
On what day would you like to work out? Type 'Monday', 'Tuesday', etc.: Saturday
What is your schedule like on this day?
Enter name of an event (or 'Q' to exit): PSTAT 170
When does 'PSTAT 170' start? Type an hour between 0 and 23: 11
When does 'PSTAT 170' end? Enter hour after start time (max 23): 12
Now you will input this event's importance in your schedule.
How important is the commitment 'PSTAT 170' (1 to 5, 5 being highest priority): 2
Enter name of an event (or 'Q' to exit): Q
Please check out the opening times in the Rec Cen website before typing.
When is the earliest hour you are willing to work out? Type a number: 10
When is the latest hour you are willing to work out until? Type a number: 19
How many hours will your workout be? Type a number: 1
Best time: Saturday 6:00 PM
```