Title: BIO47 Database

Version: 2022.2

Last Modified: 2021-5-25

Database Maintainer: Jesse Miller, jedmill@stanford.edu

Database Creator: Jes Coyle, jes.r.coyle@gmail.com

Table of Contents

Overview

This database contains data on Sticky Monkeyflower (*Mimulus aurantiacus*) plants at the Jasper Ridge Biological Preserve (JRBP, Stanford, California) and the microbial communities inhabiting the nectar of their flowers. It also contains temperature and canopy cover data at each plant's location in most years.

The data were collected by students in the Stanford University spring quarter class, "Research in Ecology and Evolution" from 2012 - 2018 (course number BIO 44Y in 2012-2016, BIO 47 in 2017-2018). Teams of

students were each assigned to measure 6-7 plants adjacent to trails at JRBP, with multiple teams surveying each plant each year. In most years, staff collected fresh leaves from each plant prior to the start of the course for nutrient and carbon isotope analysis. Students measured plant size during the first week of the quarter and then returned to the plants during week 4 - 8 to record flower production and the proportion of flowers visited by pollinators. Part-way through the quarter (usually week 6), students tagged unopened flowers and then either left them exposed or enclosed them in a cage (to exclude hummingbirds) or a mesh bag (to exclude all pollinators). After the flowers opened, students harvested the flowers to analyze the chemistry and microbial communities present in the nectar. Microbial abundance was assessed by plating nectar dilutions on agar plates and counting colonies. In some years (2016-2018) the species present in nectar microbe communities was assessed by sequencing a selection of these cultured colonies. In other years the composition of nectar microbe communities were assessed via culture-independent methods.

Missing data always denoted with NA or a blank.

Database Structure

There are two versions of the BIO47 Database:

Complete BIO47 Database

The Complete BIO47 Database contains all of the original data files from 2012 - 2018 as well as all of the code and intermediate tables used to compile the multi-year aggregated database tables. The complete database is stored on the Fukami Lab Stanford Box folder and has the following folder architecture:

aggregated/

- main_tables/
  - Contains the main database tables. These are the data that should be used for analysis.
- student_tables/
  - Contains any derived or additional tables created from the main database tables that were distributed to students.

cleaning/

- Contains intermediate tables needed before the main tables could be compiled. Generated by R scripts in the code folder. Most users will not need to view tables in this folder.

code/

- R scripts used to compile the data across years. Most users will not need to view the contents of this folder.

raw/

- Contains the original data as it was collected each year. Organized by year.

Each year, the database maintainer collects and compiles data generated by the class using their own organizational system. These “raw” data are saved in the raw folder. To update the database each year, the database maintainer then adds these data to the existing database tables in the aggregated/main_tables folder and distributes these data to the students as a zipped folder (see below) in order for students to conduct their final projects. The database maintainer may save scripts used to aggregate data in the code folder and intermediate tables in the cleaning folder. See the BIO47 Database

Compilation Notes document for information about how the 2012-2017 data were originally compiled into this database.

Users intending to analyze these data should only use tables in the aggregated/main_tables folder. Most users will not need to access any of the other folders.

Student BIO47 Database

The Student BIO47 Database is stored and distributed in a folder named BIO47_Data that is zipped and distributed to students. Students should receive a copy of this database at the beginning of the course that contains prior years' data so that they can practice data analysis in R in the Prelab assignments. The database maintainer will distribute an updated version of the Student Database later in the quarter that contains the current years data once it is available.

This folder contains an abbreviated version of the database documentation (BIO47_Data_Student_README.pdf, Table Descriptions.csv), instructions for using the database (BIO47 Database Instructions.pdf), an R Project file to allow easy access to files in the database (BIO47_Data.Rproj) and three main folders:

code/

A folder with R scripts used by the current BIO47 class and an application ("app") for creating summary tables from the database (make_analysis_table)

data/

main_analysis/

A folder containing the primary database. Files in this folder should not be changed by anyone except the database maintainer. For the make_analysis_table application to work, this folder must contain: flowers.csv, flower_counts.csv, plant_measurements.csv, plants.csv, temperature_summary_long.csv, week_dates.csv, nectar_community.csv, otu_names.csv. Including ibuttons.csv is not necessary, but some students may wish to utilize these raw data.

analysis_tables/

A folder that can contain any pre-formatted data tables that students will need in the exercises that they complete in the first few weeks of class. Users can should save their own tables in this folder.

output/

A folder where any figures or other content generated in R should be saved.

Important Note: The make_analysis_table app requires that the BIO47_Data have the file "Table Descriptions.csv" in the main directory in order to properly display variable names.

Users should not change any of the files that are in this database. Instead, users should save their tables for analysis in the data/analysis_tables folder and save any other content in the output folder.

Note to database maintainer: Whenever you make substantial changes to the data in this folder, increment the database version in this document. The version number should change each year to reflect the fact that new data are added by the class each year.

Data Acquisition and Quality Assurance

In the following instructions, “you” is directed at the database maintainer.

Web-form Data Collection

Students in BIO47 work in pairs (and sometimes groups of three) to collect data in lab each week. In order to reduce data entry errors, we require that each partner enter data online separately. You, the database maintainer, should check whether 1) whether every student as entered data and 2) whether the data from students in the same team match. Once you ensure the quality of all data you should create a single data table without duplicate observations. If data from the same team do not match, the you must contact the students for data re-entry until the data are correct. Students receive a Post-lab grade for whether they correctly enter data and it is your job to enter these grades on Canvas.

The database maintainer may use any system for collecting data from students. One system that has worked well is to use the Stanford Google Drive and collect data using Google Forms. You (the database maintainer) should create a Google Form each week for each table that is listed in the lab protocol. You then need to include a link to each of the necessary forms in the Post-lab Data Entry Assignment on Canvas for each week. You can find the link to the form by clicking the “Send” button at the top of the form and then copying the link that appears when you click the link icon. Students are required to enter data by 10pm the same day that they collect data in their lab section. Because the first lab section is typically on Tuesday, you will need to make sure to create the data entry form(s) and put the link(s) in the Canvas Post-lab assignment by Monday night.

To reduce data entry errors, the fields in the form should match and be in the same order as the columns in the table. Under the Settings for each form you can also set the form to automatically collect email addresses and restrict responses to Stanford University users. By clicking on the Responses tab, you should set the form to save responses to a Google Sheet by clicking the three vertical dots icon and then selecting “Select response destination”. Then, as students enter data their responses will be saved to a continuously updating Google Sheet that includes a timestamp.

Timestamps are a useful way to allow students to re-enter previously incorrect data because they allow you to sort the form by time and only keep the most recent data for each observation. You should also create a field at the top of the form for students to enter their team and lab section. This will allow you to quickly sort data from the same teams.

Google Forms will allows response validation for each field, which allows you to designate the types of responses that are allowed. For example, if you are requesting data on plant height in centimeters, you can set the field to only accept numbers between 10 and 200, which ensures that students do not accidentally enter the height in meters. You can also designate regular expressions to control the data entered in text-based fields. However, whenever you require a test-based response (e.g. Team) it is best to create a multiple-choice field so that responses from students on the same team are easier to automatically match up.

Be sure to give the students clear written instructions on how to handle missing data. The best way is to instruct them to leave missing fields blank and not enter NA.

Team Organization

There are usually 4-5 section of BIO47 taught in the spring. The sections will have numbers, but are usually referred to by their lab time (e.g. Tue AM or Wed PM). In each section there are up to 10 teams of students, each assigned a color (which refers to the lab table they sit at) and a shape (square or circle) to differentiate the two teams which are assigned the same color:

1A	Orange circle	1B	Orange square
2A	Blue circle	2B	Blue square

3A	Pink circle	3B	Pink square
4A	Yellow circle	4B	Yellow square
5A	Green circle	5B	Green square

Each color-shape group is assigned to measure 5 - 7 marked and numbered *Mimulus* plants at Jasper Ridge. The locations and numbers of these plants are on a map of Jasper Ridge in which the locations of the plants are marked with circles/squares that match the color and shape of the group to which they are assigned. These assignments change each year. At least one group will be assigned to only measure plants along cart-accessible roads, in case there are students who need mobility accommodation.

Because the same color-shape team may exist in different sections, most *Mimulus* plants will be measured by multiple teams from different sections each week. However, within a given section, plants are generally not sampled by different teams. In addition, since there are generally fewer than 20 students in a section, not all color-shape teams will exist in each section. Note that students sometimes need to make up a missed lab and will attend a lab section that is different from their usual section. In this case, students may be entering data with a team that is not their usual team and you should be aware of aby absences/makeup labs when checking for data entry compliance and matching up data within teams.

Data Collection Timeline

Students will collect the following data throughout the course:

- An initial survey of plant size and flowering status in week 1
- Weekly surveys of flowering and pollination, typically during weeks 4 - 8
- Data on flowers collected for analysis of microbial communities, typically during week 5 or 6
- Data on microbial abundance and diversity within these flowers determined based on counts of colonies and morphotypes from plated nectar, acquired one week after nectar is plated
- Data on OTU presence and diversity based on rDNA 28S (fungi) and 16S (bacteria) sequences, typically available two weeks after nectar is plated

Students will enter the plant data and weekly flower count data the same day that they are collected. However, because the nectar microbes require a few days to incubate before their colonies can be counted, the flower-level data on nectar and microbial abundance will be entered the week after students collect the flowers and plate the nectar. DNA sequencing data is not available until two weeks after the nectar is plated because students must wait one week for the colonies to grow before they can perform PCR on the colonies and then need an additional week to analyze the DNA sequences.

The following is a typical timeline for when different types of data are generated during the course; you should check with the course instructors for the actual timeline.

Week	Lab activity	Data Entered
1	Visit Jasper Ridge and record initial plant measurements	Plant dimensions and presence of open flowers.
2	Collect practice flowers near Gilbert Hall to practice nectar extraction and plating. Students record flower and nectar data, but do not enter until the following week.	
3	Count microbe colonies on practice flower plates.	Flower, nectar and microbe data from practice flowers plated in week 2.

	Set up artificial flower experiment at Stock Farm.	
4	Measure nectar volume from artificial flower experiment. Weekly flower counts at Jasper Ridge.	Nectar volume from artificial flowers. Number of open flowers and flowers with open/closed stigmas for JR plants.
5	Weekly flower counts at Jasper Ridge. Tag unopened flower buds for collection in week 6.	Number of open flowers and flowers with open/closed stigmas for JR plants.
6	Weekly flower counts at Jasper Ridge. Collect tagged flowers, extract and plate nectar.	Number of open flowers and flowers with open/closed stigmas for JR plants.
7	Count microbe colonies from flower nectar plated in week 6. PCR and sequencing of selected colonies on plates.	Flower, nectar and microbe data from JR flower nectar plated in week 6.
8	Analysis of DNA sequence data to assign taxonomic identifications to colonies. Download ibutton temperature data. Weekly flower counts at Jasper Ridge.	Taxon IDs of colonies sequenced in week 7. CSV files with ibutton temperature records from each plant. Number of open flowers and flowers with open/closed stigmas for JR plants.
9	Weekly flower counts at Jasper Ridge.	Number of open flowers and flowers with open/closed stigmas for JR plants.

Note that the data collected during week 1 and 2 are for students to practice collecting nectar from flowers on campus and will not end up in the database. There may be other experiments for which the students will need to enter data, but which are not part of the core database. You should coordinate with the instructors for how the want these data aggregated and distributed to students.

Although students may be collecting and entering data through week 9, they will need a final version of the database by Friday of week 8 in order to complete their data analyses for their projects. This leaves very little time for you to collect, quality check and aggregate the data. Discuss with the instructors the best way to handle these deadlines. We, the former database maintainers, strongly recommend that you aggregate the data as they come in each week and test your data aggregation scripts on “fake” data prior to when the actual data are available. Even after running your scripts to aggregate the final data, it is a good idea to open the updated database tables in Excel and manually sort them to look for duplicate rows or unlikely values.

How to Update the Database

Workflow Summary

During the quarter, the database maintainer should utilize their own organizational system to acquire, compile and quality check data from students. These data include initial plant measurements from week 1, weekly surveys of flowering and pollination, data on flowers collected for analysis of microbial communities, and nectar microbial community data. The database maintainer should also request (or will receive) data on monthly PAR and delta C13 for each plant from the Biology Teaching Labs manager as well as csv files with ibutton temperature measurements downloaded by students.

Once all of these intermediate files are prepared, these data should be incorporated into the BIO47 Database by following the steps summarized here. Please read the detailed instructions in the subsequent sections before altering the database. Note that the values of some columns in the database tables are dependent on data in other tables. We recommend that the database maintainer update the tables in the following order to avoid re-calculating tables (details for these steps appear in the subsections below):

Step 1: Quantify nectar microbe communities in flowers.

a. Aggregate and standardize the sequencing data and update **otu_names** with any new taxa.
b. Update the **nectar_community** table with the presence/absence of each OTU found in each flower.
c. Calculate microbial diversity and combine it with flower-level measurements to update the **flowers** table.

Step 2: Summarize weekly flower count data and update the **flower_counts** table.

Step 3: Summarize and aggregate plant-level measurements and update the **plant_measurements** and **plants** tables.

a. Calculate total microbial OTUs for each plant from the flowers table.
b. Calculate the week when each plant was first observed flowering using the flower_counts table and initial week 1 data.
c. Calculate plant size measurements from the week 1 data.
d. Compile the PAR and delta C13 data.
e. Aggregate data from steps a - e and update the plant_measurements table.
f. Update the plants table to indicate which plants were measured this year and, for those plants, increment the number of years sampled by one year.

Step 4: Aggregate and summarize ibutton temperature data.

a. Update the **week_dates** table by adding the calendar dates for the beginning and end of each week of the quarter as well as any other time periods over which temperature data should be summarized.
b. Update the **ibuttons** table with ibutton temperature records from each plant. These data should only contain measurements during the time period when ibuttons were actively deployed at plants.
c. Summarize ibutton measurements for each plant within time periods specified in the week_dates table. Add these summaries to **temperature_summary_long** and **temperature_summary_wide**.

Detailed Instructions

Step 1: Flower-level Data and Nectar Microbe Communities

a. Aggregate and standardize the sequencing data and update otu_names with any new taxa.

The otu_names table contains one row for each unique OTU identified *across all years*. Determine all of the unique OTUs present in this year's data and determine whether any are the same as in prior years. We expect that most OTUs should be the same as prior years since most OTUs are only identified to the Genus level. When in doubt, be conservative with taxon assignment and do not create a new species-level OTU unless you are sure of the identification. If there are any new OTUS, give each of them a unique ID by following the naming convention in the OTU_ID column of the otu_names table. OTU_IDs consist of either FOTU (for yeast) or BOTU (for bacteria) followed by a number, with no spaces or characters in between. If you do not follow this convention, the make_data_tables app will not work correctly. Sort the otu_names table by the OTU_ID column to be sure that you did not duplicate an existing OTU_ID.

Students often isolate microbial contaminants that are not members of the nectar microbial community, such as human commensal bacteria (e.g. E. coli). Do not include these taxa in the database.

b. Update the nectar_community table with the presence/absence of each OTU found in each flower.

The nectar_community table contains one row for each unique OTU in each flower from a given plant in a given year. If there are two rows with the same OTU from the same flower, the make_analysis_table app will malfunction. Tabulate which OTUs are present in each flower of this year's data and then add a row to the nectar_community table for each OTU in each flower. For example, if flower 398 has OTUs BOTU1, BOTU3, and FOTU present, you would add three rows to the nectar_community table for flower 398. OTU_IDs must match the OTU_IDs listed in the otu_names table or the make_analysis_table app will malfunction.

Students may have collected data on the number of colonies with different morphotypes found in each flower. If so, use the match each morphotype from each flower with its OTU_ID and then use the number of colonies counted for each morphotype and the nectar volume plated to calculate CFU per uL. Include these cell densities in the CFU_per_uL column in the nectar_community table.

c. Calculate microbial diversity and combine it with flower-level measurements to update the flowers table.

Tabulate the number of unique bacterial and fungal OTUs and morphotypes in each flower. Calculate the nectar volume (in uL) extracted from each flower using the length of the nectar column in the microcapillary tube (reported by students) and the internal diameter of the microcapillary tube (previously 0.701 mm for the 25 uL tubes, but be sure to verify this). Alternatively, you can use the ratio of the measured length of the nectar column and the total length of the tube to calculate the fraction of the total possible volume in the tube that is filled. Calculate total cell density for bacteria and fungi using the number of colonies that students counted on each plate, the volume and dilution factor of the nectar solution plated, the volume of nectar extracted, and the volume of the saline solution into which the nectar was initially diluted. Consult the lab protocol for these volumes. The formula is:

$$\text{CFU per uL} = \frac{\text{\# colonies} \times (\text{nectar uL} + \text{saline uL}) \times \text{dilution factor}}{\text{plated uL} \times \text{nectar uL}}$$

Note: if nectar volume is zero, use 0.2 uL instead in the denominator so that cell density is not infinite. Dilution factor should be a number greater than 1. For example, if students plated a 1:100 dilution the dilution factor is 100.

Combine these calculated data with the flower numbers that identify a unique flower in a given year and whether each flower was caged, bagged and had an open or closed stigma. Be sure to match the format of the data already entered in the flowers data table. For example, stigma status is recorded in the Stigma column as CLOSED or OPEN. Deviations from the existing format will cause the make_data_table app to malfunction. Use NA for any data not collected or missing.

Step 2: Weekly Flower Counts

Each week during approximately weeks 4 - 9 multiple teams will count the number of flowers on each plant at Jasper Ridge and then randomly survey 12 flowers, recording how many have open versus closed stigmas. The flower_counts table contains one row for each plant each week in each year and reports the average number of flowers counted by all teams across the different sections as well as the average fraction of flowers with open stigmas. Sometimes plants will have fewer than 12 open flowers (and sometimes students do not follow the directions to count stigmas from only 12 flowers), therefore, first calculate the fraction of close stigma flowers from each team before averaging across teams. Also calculate and record the average number of open and closed stigma flowers across teams, even though these data are not really useful.

Step 3: Plant-level Data

a. Calculate total microbial OTUs for each plant from the flowers table.

   Tabulate the number of unique bacterial and fungal OTUs found in flowers from each plant. These data will become the BOTU_Richness and FOTU_Richness columns in the plant_measurements table.

b. Calculate the week when each plant was first observed flowering using the flower_counts table and initial week 1 data.

   For each plant that did not have flowers in the first week, determine the earliest week when flowers were observed in the flower_counts table. If flowers were never counted record NA. Then determine the calendar date that corresponds to the Monday of the week when flowers were first observed. You may want to update the week_dates table (see step 4a) and then use this table to automatically populate these dates. These data will become the Week_flowers_present and Date_flowers_present columns in the plant_measurements table.

c. Calculate plant size measurements from the week 1 data.

   During week 1, multiple teams will record the length, width and depth of a rectangular prism that encompasses each plant. Calculate the volume of this prism (in m^3) measured by each team. Then calculate the average volume and height (in m) for each plant across teams.

d. Compile the PAR and delta C13 data.

   Obtain monthly PAR data (mol m^{-2} day^{-1}) from Nona Chiariello, based on her analysis of canopy photographs above each plant. If new data are not available you may need to use data from prior years. If so, be sure to record which year was used in this README file in the detailed description of the plant_measurements table (in the Main Table Descriptions section). Obtain delta C13 data from the staff.

e. Aggregate data from steps a - e and update the plant_measurements table.

f. Update the plants table to indicate which plants were measured this year and, for those plants, increment the number of years sampled by one year.

Step 4: Temperature Data

a. Update the week_dates table by adding the calendar dates for the beginning and end of each week of the quarter as well as any other time periods over which temperature data should be summarized.

   Be sure to follow exactly the same formatting that is in the existing week_dates table. For each week, include the month during which that week occurs (Apr or May). The make_data_tables app uses this information to determine which weeks occur in which months.

b. Update the ibuttons table with ibutton temperature records from each plant. These data should only contain measurements during the time period when ibuttons were actively deployed at plants.

   Students will download data from ibuttons with 1-wire viewer and save as csv files. These files should be named with the plant number only (e.g. the data from plant 99 would be named 99.csv). Put all csv files in a folder named with the year in the cleaning/ibuttons folder. Write a METADATA text file (e.g. 2018_ibutton_METADATA.txt) for the ibuttons including information on the dates when ibuttons were deployed. Include the dates between which all ibuttons were deployed outdoors on plants. See the cleaning/ibuttons folder for examples from prior years.

   Open aggregate_ibuttons_allyear.R from code folder. Add new section for the year that will combine all files from the folder into a single csv file and remove measurements that were outside the time window when ibuttons were deployed. Do this by copying example code from prior year and updating the start and end dates and folder names. Running this code will create an intermediate table in the cleaning folder named ibuttons_YEAR.csv. Note that you will need to change the working directory at the top of the script to point to the BIO47 Database folder location on your computer.

   Add the cleaned ibutton data from this year to the ibuttons table by running the section of the aggregate_ibuttons_allyears.R script that begins with the line reading: `### Combine all ibutton data into a single file`

c. Summarize ibutton measurements for each plant within time periods specified in the week_dates table. Add these summaries to temperature_summary_long and temperature_summary_wide.

   Run the rest of the aggregate_ibuttons_allyear.R script starting from the section titled: `### Summarize ibutton data` to the end of the script. This will combine all of the prior years' data with the new data and calculate summary statistics in the temp_summary_long and temperature_summary_wide tables. The script will calculate temperature average, minimum, maximum and range across each of the time periods that you include in the week_dates table.

   The aggregate_ibuttons_allyears.R script requires that you have installed the following packages: dplyr, tidyr, chron, lubridate. It also requires that you have the script summarize_ibuttons.R in the code folder.

Distributing the Database

At the beginning of the quarter you should distribute a version of the BIO47 Student Database containing prior years' data and any additional tables that students will need to complete the Prelab R assignments in the first four weeks of the quarter. Put any additional tables in the data/analysis_tables folder. You should also update the "BIO47 Database and App Instructions.pdf". An editable Word document corresponding to this file is in the main database folder. Do not distribute the Word document to students.

Export the file as a pdf. Upload the database along with as a zipped folder to Canvas (BIO47_Data.zip) and then include detailed instructions of how to download the folder, unzip and use the database. Examples of instructions can be found in the Canvas Prelab assignments from prior years. You can find the final BIO47_Data.zip file that was distributed to students last year in the main folder of the database on Stanford Box.

After you have compiled the final data from the current year, you should upload the new files to the Fukami Lab Stanford Box Folder. Replace the existing old table files. Do not create new files with different names.

Upload any raw data to the Stanford Box folder. The ibutton files go into the cleaning/ibuttons folder in a subfolder labeled with the year, along with the METADATA file that you created describing the dates that the ibuttons were deployed. Any DNA sequencing data should go in the raw folder in a subfolder labeled with the year.

Finally, update this database documentation as needed to reflect any changes or additional useful information. Include this updated documentation in when you update the database on Box.

Create a new version of the Student Database using the updated data. Be sure that is you made any changes to the Main Table Descriptions section of this README that you copy those changes into the corresponding section of the BIO47 Database and App Instructions.pdf that is distributed to students. Upload the new version of the zipped Student Database (BIO47_Data.zip) as well as an unzipped version with all of the updated tables so that students can download and replace tables individually. Students may want to do this if they have created new tables in their analysis_tables folder that they don’t want to be deleted by a clean re-install.

Main Table Descriptions

Table name: flower_counts.csv

Description: Each row of this table contains data on the open flowers present on a plant in a given year during a particular week of the quarter. Most plants were surveyed by multiple teams of students (on different days of the week) so flower counts and the fraction of flowers with closed stigmas are averaged across teams within a week. To estimate pollinator activity, students were instructed to randomly sample 12 flowers on each plant and determine whether stigmas were open or closed, indicated that a flower has likely been visited by a pollinator. Note that some students may not have followed these instructions or the plant may have had fewer than 12 open flowers on a given day. Therefore, an weekly average of the number of open/closed stigma flowers is likely meaningless (columns 6 and 7).

Column	Name	Description
1	Year	Year the plant was sampled.
2	Plant	Unique number identifying the plant. Corresponds to column in plants.csv.
3	Week	Week of the spring quarter when the flowers were counted.
4	Num_flowers	Number of open (stigma visible) flowers on the plant. Averaged across all teams surveying the plant in a particular week.
5	Fraction_closed_stigmas	Estimated fraction of flowers surveyed with closed stigmas. Averaged across all teams surveying the plant in a particular week.
6	Num_closed_stigmas	Number of flowers with closed stigmas out of a random sample of flowers. Averaged across all teams surveying the plant in a particular week.
7	Num_open_stigmas	Number of flowers with open stigmas out of a random sample of flowers. Averaged across all teams surveying the plant in a particular week.

Table name: flowers.csv

Description: Each row of this data table corresponds to a single flower collected from a plant at Jasper Ridge in a given year. Each flower was either enclosed in a mesh bag, enclosed in a caged to exclude hummingbirds or left exposed.

Nectar microbe communities were assessed by culturing for 7 days at 25 C. Nectar was extracted via capillary tube and pipetted into 0.85% NaCl solution and then dilutions of this nectar solution spread on agar plates. Different nectar dilutions were used in different years, so variables that are not corrected for dilution may not be comparable across years (i.e. number of colony morphotypes and number of OTUs).

2018: Original dilution of nectar in 135 uL saline solution. Yeast plated at the original dilution and bacteria plated at 1:10 dilution.

2017: Original dilution of nectar in 40 uL saline solution. Yeast plated at 1:10 dilution and bacteria plated at 1:100 dilution.

R2A media with cyclohexamide was used to selectively culture bacteria of the original nectar solution, whereas YM media with chloramphenicol was sued to selectively culture yeast. Bacterial and yeast cell densities were then estimated from colony counts on the agar plates. The original nectar solution was used for all nectar chemistry analyses.

Several different methods were used in different years to estimate microbial diversity. In 2012 and 2016 - 2018, students selected colonies from their plates, amplified and sequenced either the ITS region (yeast) or 16S region (bacteria) of the ribosomal RNA gene, then BLASTed resulting sequences against GenBank to determine unique OTUs. In 2014, nectar was sequenced using _____ and microbial OTU richness estimated using the Chao1 richness estimator. In 2013, _____.

Note that several variables were only measured in some years. Flower data are missing for 2015 because that years students collected nectar from a different plant species.

Column	Name	Description
1	Plant	Unique number identifying the plant. Corresponds to column in plants.csv.
2	Year	Year the plant was sampled.
3	Week	Week of the spring quarter when the flower was collected.
4	Flower	Number identifying a flower within a particular year.
5	Bagged	Is TRUE if the flower was enclosed in a mesh bag prior to opening. Is FALSE otherwise.
6	Caged	Is TRUE if the flower was enclosed in a mesh bag prior to opening. Is FALSE otherwise.
7	Stigma	Is OPEN or CLOSED depending on whether the stigma was observed as open or closed prior to harvest.

8	Age_category	Age of the flower at harvest based on the color and appearance of the anthers. 1 = both sets of anthers yellow 2 = one set of anthers orange 3 = both sets of anthers orange 4 = anthers dark and wilted
9	Age_days	Number of days after opening when the flower was harvested .
10	Nectar_pH	pH of the flower nectar after dilution in 40 uL of 0.85% NaCl. Measured using a pH strip.
11	Nectar_uL	Volume (uL) of nectar extracted from the flower. Measured via microcapillary tube.
12	EtOH_Percent	Ethanol percentage of the nectar.
13	Fructose_mg_per_mL	Fructose concentration of the nectar (mg / mL).
14	Glucose_mg_per_mL	Glucose concentration of the nectar (mg / mL).
15	Sucrose_mg_per_mL	Sucrose concentration of the nectar (mg / mL).
16	Bacterial_CFU_per_uL	Density of bacterial cells in the nectar (CFU / uL).
17	Bacterial_morphotypes	Number of different bacterial colony morphotypes observed on a plate. Not corrected for dilution.
18	Num_BOTU	Number of bacterial OTUs present. See details above for methods.
19	Fungal_CFU_per_uL	Density of yeast cells in the nectar (CFU / uL).
20	Fungal_morphotypes	Number of different yeast colony morphotypes observed on a plate. Obvious contaminants (molds) were excluded. Not corrected for dilution.
21	Num_FOTU	Number of yeast OTUs present. See details above for methods.

Table name: nectar_community.csv

Description: Each row in this table corresponds to an OTU observed in the nectar of a flower collected from a plant in a given year. OTU identities were determined by different methods in different years.

In 2012, 2016 - 2018, students BLASTed DNA sequences against GenBank and chose the top result to assign an OTU identity to a sequence.

In 2014,

In 2013,

Column	Name	Description
1	Plant	Unique number identifying the plant. Corresponds to column in plants.csv.
2	Year	Year the plant was sampled.
3	Flower	Number identifying a flower within a particular year. Corresponds to column in flowers.csv.
4	OTU_ID	Number identifying a unique OTU. Corresponds to column in otu_names.csv.
5	CFU_per_uL	Number of CFUs per uL of a given OTU counted for a given flower. Calculated by matching morphotypes to identificed DNA sequences. Data not currently available.

Table name: otu_names.csv

Description: This table gives the taxonomic identities of OTUs observed in the nectar of flowers. OTU_IDs are unique across years to enable a comparison of microbial community composition across years.

Column	Name	Description
1	OTU_ID	Unique code identifying an OTU. Matches code in nectar_community table.
2	Binomial_name	Latin binomial name identifying the taxon to which the OTU belongs. Usually uses the lowest known taxon to which the species belongs.
3	Group	Is YEAST or BACTERIA depending on whether the OTU is a yeast or bacteria.
4	Class	Class to which the OTU belongs (if known).
5	Order	Order to which the OTU belongs (if known).
6	Family	Family to which the OTU belongs (if known).
7	Genus	Genus to which the OTU belongs (if known).
8	Species	Species to which the OTU belongs (if known).

Table name: **plant_measurements.csv**

Description: Each row in this table corresponds to measurements of a particular plant in a particular year. Students measured plant size using measuring tapes during the first week of the quarter. In most years, staff collected fresh leaves from each plant within one month prior to the start of the spring quarter for nutrient and carbon isotope analysis.

Note: Plant measurement data are missing from 2013, but rows are included for these plants as a placeholder if the data are found.

Column	Name	Description
1	Plant	Unique number identifying the plant. Corresponds to column in plants.csv.
2	Year	Year the plant was measured.
3	Volume_m3	Volume of the plant in cubic meters. Calculated as the volume of a rectangular prism enclosing the entire plant based on height, width and depth measurements.
4	Height_m	Height of the plant in meters.
5-17	[Jan-Dec]_PAR_mol_m2_day	Estimated total photosynthetically active radiation (PAR) received by the plant in a month prior to the spring quarter. PAR was estimated from hemispherical digital photographs capturing canopy cover above the plant using the Gap Light Analyzer software. Units are moles per square meter per day. Data from 2018 are identical and are from photos taken in 2017. Data from 2016 and 2015 are identical and are from photos taken in 2015.
18	deltaC13	Ratio of C13 to C12 in plant leaf tissue (parts per thousand).
19	Foliar_N_percent	Percentage of leaf tissue comprised of nitrogen.
20	Foliar_C_percent	Percentage of leaf tissue comprised of carbon.
21	Watered	Is YES if a plant was watered during the spring quarter. Is NO otherwise.
22	BOTU_Richness	Total number of bacterial operational taxonomic units (OTU) on a plant based on sequencing data.
23	FOTU_Richness	Total number of yeast operational taxonomic units (OTU) on a plant based on sequencing data.

24	Week_flowers_present	First week that students observed flowers on a plant. These dates were taken from the original plants data for years 2016 - 2017 and calculated for years 2012 - 2015 from the flower_counts.csv table. Because flower counts usually did not start until week 4, flowers could have been observed earlier in 2016 - 2017 than in previous years. In a few cases, the week when flowers were first observed in the flower counts data was earlier than recorded in the plants data. When this occurred, the earliest week of first flowering was used.
25	Date_flowers_present	Date of the first day of the week that students first observed flowers on a plant.

Table name: plants.csv

Description: Each row of this table corresponds to a plant in the database. The table contains information about each plant's location (UTM coordinates) and that it was measured.

Column	Name	Description
1	Plant	Unique number identifying the plant.
2	First_year	First yeas that a plant was measured.
3	Last_year	Most recent year that a plant was measured.
4	Num_years	Number of years that a plant was measured.
5	UTM_N	North UTM coordinate of the plant (zone 10).
6	UTM_E	East UTM coordinate of the plant (zone 10).

Table name: temperature_summary_long.csv

Description: This table contains temperature data collected by iButtons located at each plant. iButtons recorded temperature in Celcius every 10-20 minutes and these data were then summarized into daily temperature statistics (mean, max, min, range). Each row in this table contains a daily temperature summary at plant averaged over a particular time period in a particular year. For example, if the Time_period is “Week3”, then the Temp_daily_max column gives the average maximum daily temperature during the third week of the quarter. See the week_dates.csv table for the specific calendar dates that corresponds to spring quarter weeks in each year.

Raw iButtons files were missing from 2014 and 2015. Data from these years are based on summary data provided.

Column	Name	Description
1	Year	Year that temperature was measured.
2	Plant	Unique number identifying the plant. Corresponds to column in plants.csv.
3	Time_period	Period of time over which daily temperature summaries were averaged. See week_dates.csv for the calendar dates of each spring quarter week each year. Monthly time periods correspond to the actual calendar month.
4	Temp_daily_max	Average daily maximum temperature (C) over the given time period in the given year.
5	Temp_daily_mean	Average daily mean temperature (C) over the given time period in the given year.
6	Temp_daily_min	Average daily minimum temperature (C) over the given time period in the given year.
7	Temp_daily_range	Average daily temperature range (C) over the given time period in the given year.

Table name: temperature_summary_wide.csv

Description: Data in this table are identical to those in temperature_summary_long.csv. The only difference is that each row corresponds to a plant in a particular year and each column is a daily temperature summary averaged over a given time period. For example, the column Week3_Temp_daily_max gives the average maximum daily temperature during the third week of the quarter.