Ask

Task:

- Analyze smart device usage data in order to gain insight into how consumers use non-Bellabeat smart devices.
- Use current market trends to guide Bellabeat's decisions for future products or systems.

Stakeholders:

- Bellabeat marketing department
- Urška Sršen, cofounder and Chief Creative Officer
- Sando Mur, cofounder and executive

Prepare

Public Kaggle dataset used as primary data source per client suggestion:
https://www.kaggle.com/datasets/arashnic/fitbit

Focused on the data for dailyActivity, dailyCalories, dailyIntensities, dailySteps, sleepDay, and weightLogInfo to get a clearer picture on macro level habits of FitBit users.

Potential limitations with this dataset:

- Number of participants
- Self report data
- Unknown sex of participants. As Bellabeat markets their products to women, the FitBit dataset may not accurately represent their target market.

Process

CSVs stored on Google Sheets to verify and clean data. Found the dailyCalories, dailyIntensities, and dailySteps data to be redundant with the dailyActivity sheet and cut them. Converted the Date column for the sleepDay and weightLogInfo sheets to datetime for easier analysis. Clean data then uploaded to BigQuery for processing and analysis.

Queried total users per table:

```
SELECT
  DISTINCT Id
FROM
  `bellabeat-case-study-370423.FitBit_public.daily_activity`;

SELECT
  DISTINCT Id
FROM
  `bellabeat-case-study-370423.FitBit_public.daily_sleep`;

SELECT
  DISTINCT Id
FROM
  `bellabeat-case-study-370423.FitBit_public.weight_log`;
```

Results:

- daily_activity = 33 unique IDs
- daily_sleep = 24 unique IDs
- weight_log = 8 unique IDs

There are clear inconsistencies in the number of users across the three tables, though this may be the result of users opting not to use different features of the app. To verify that the user IDs were consistent across all tables, we can join them by ID:

```
SELECT
  A.Id,
  AVG(A.TotalDistance) AS avg_distance,
  AVG(A.TotalSteps) AS avg_steps,
  AVG(A.LoggedActivitiesDistance) AS avg_logged_activity,
  AVG(A.Calories) AS avg_calories,
  AVG(B.TotalMinutesAsleep)/60 AS avg_sleep_hours,
  AVG(C.WeightPounds) AS avg_logged_weight
FROM
  `bellabeat-case-study-370423.FitBit_public.daily_activity` AS A
  LEFT JOIN
    `bellabeat-case-study-370423.FitBit_public.daily_sleep` AS B ON A.Id = B.Id
  LEFT JOIN
    `bellabeat-case-study-370423.FitBit_public.weight_log` AS C ON A.Id = C.Id
GROUP BY
  A.Id;
```

The results of this query are saved as a new table named “feature_usage.”

Results:

- 33 unique IDs consistent across tables
- Null values represent features not being used by the user

It is worth noting that although we have 33 unique users in this dataset, the language of the source suggests that there should only be 30 users. The reason for this discrepancy is unclear, but we will proceed with the data we have.

Analyze

While most of the data in the dataset is automatically generated by FitBit, we can determine which features are most used by analyzing the data that requires user input. These features are logged activity, calories, sleep time, and weight and have been averaged in the new “feature_usage” table.

Logged activity in the new table returns a value of 0 when that user did not enter any data for this feature. We can run the following query to check for any users with a value greater than 0:

```
SELECT
  Id
FROM
  `bellabeat-case-study-370423.FitBit_public.feature_usage`
WHERE
  avg_logged_activity > 0;
```

Result: 4 users

Calories in the new table also returns a value of 0 when that user did not enter any data for this feature. We can tell visually that of the 33 rows in our “feature_usage” table, none of them are 0, but we could run the following query if we had significantly more users or to eliminate the chance of human error:

```
SELECT
  Id
FROM
  `bellabeat-case-study-370423.FitBit_public.feature_usage`
WHERE
  avg_calories > 0;
```

Result: 33 users

Sleep time in the new table returns a null value when that user did not enter any data for this feature. We can run the following query to check for any users with a non-null value:

```
SELECT
  Id
FROM
  `bellabeat-case-study-370423.FitBit_public.feature_usage`
WHERE
  avg_sleep_hours IS NOT NULL;
```

Result: 24 users

Weight in the new table also returns a null value when that user did not enter any data for this feature. We can adjust the previous query to check for non-null values:

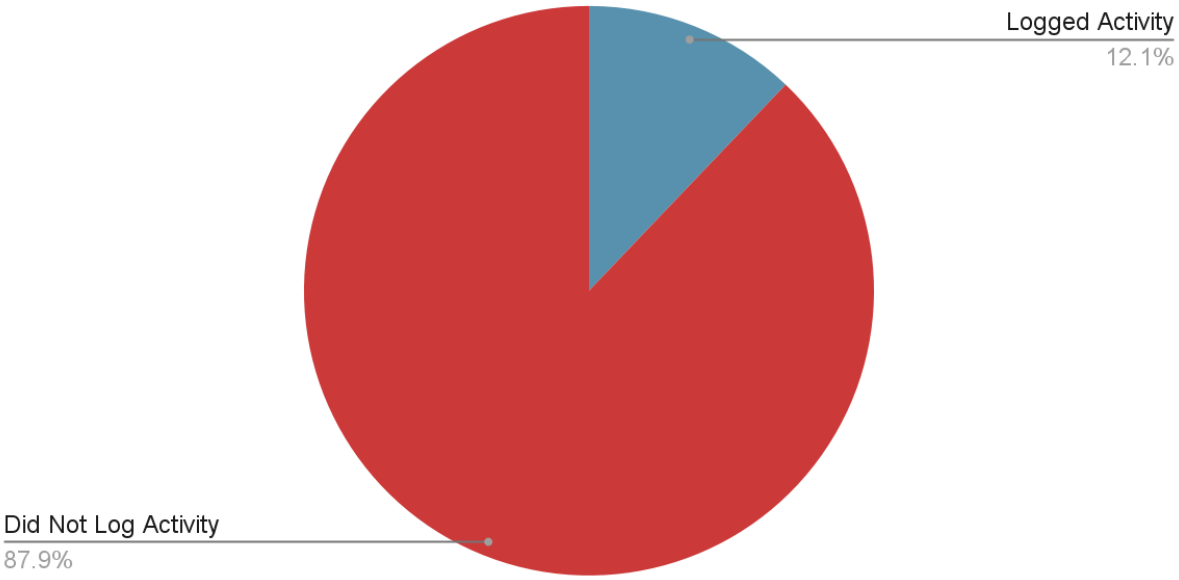
```
SELECT
  Id
FROM
  `bellabeat-case-study-370423.FitBit_public.feature_usage`
WHERE
  avg_logged_weight IS NOT NULL;
```

Result: 8 users

Share

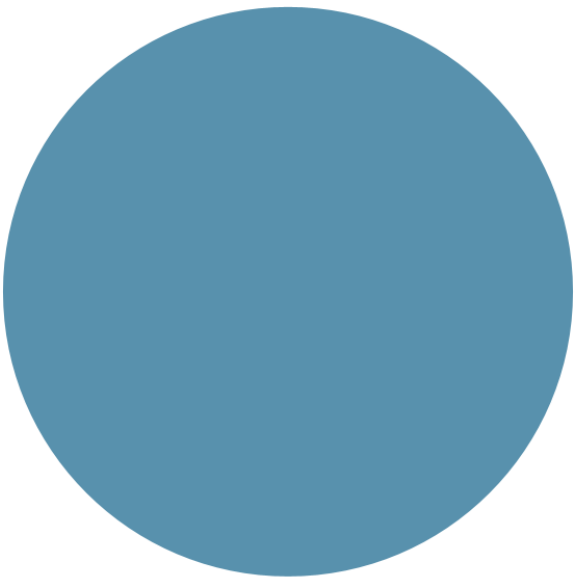
Activity

~12% of users logged activity



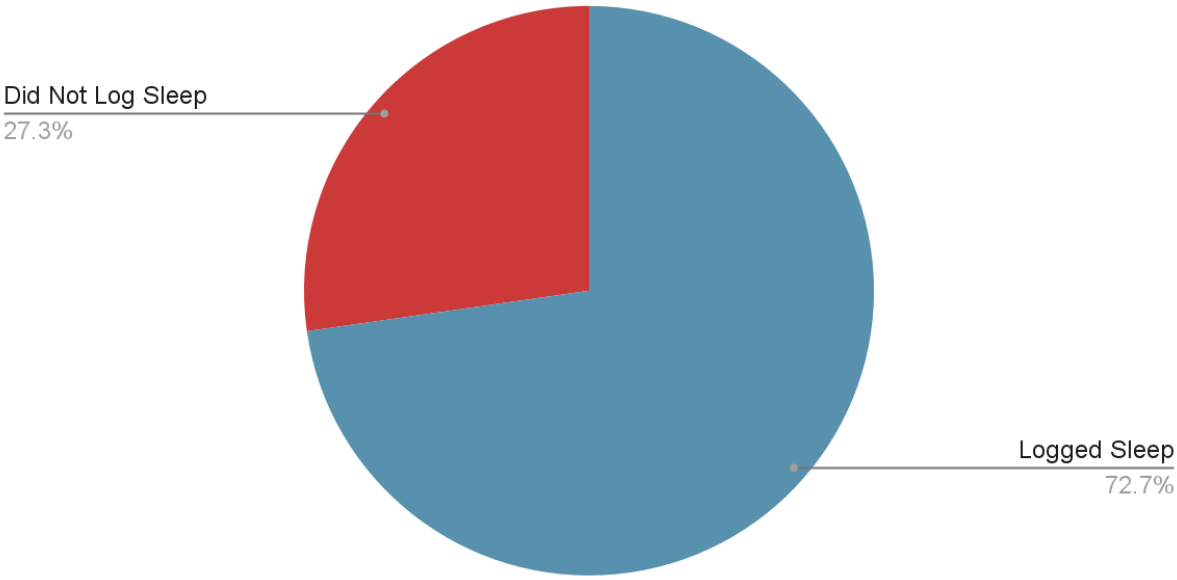
Calories

100% of users logged calories



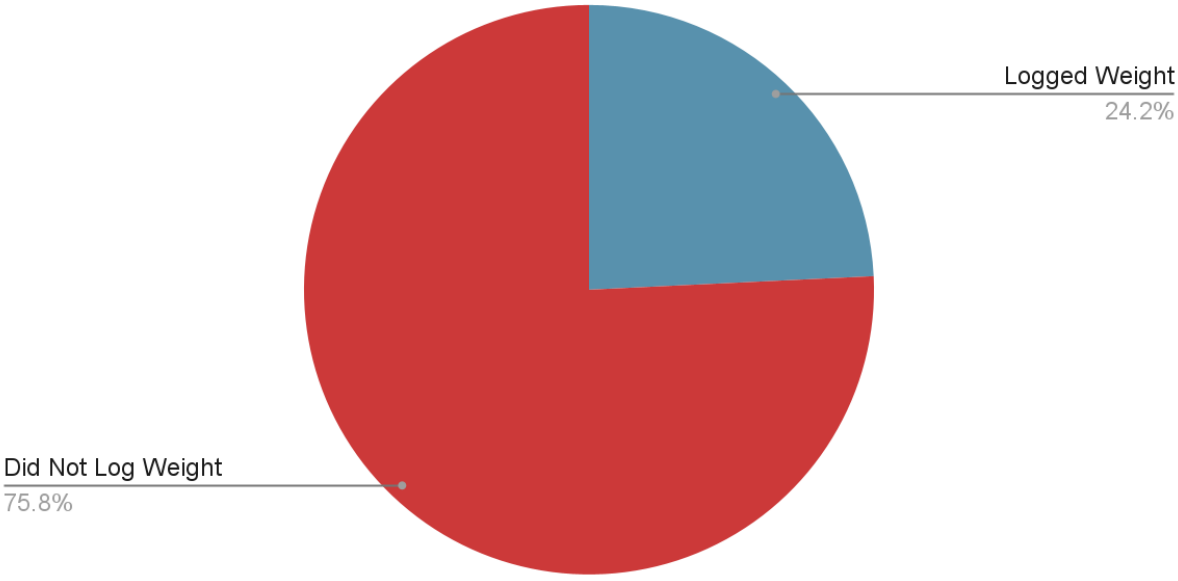
Sleep

~73% of users logged sleep



Weight

~24% of users logged weight



We can see that of the 33 users who responded to the survey, all of them used the calorie logging feature at least once. The next most used feature is sleep logging at ~73%, followed by weight at ~24% and activity at ~12%.

Act

This data is a good starting point to see which features of FitBit's app are being used, but further investigation is required to determine why some features are used more than others before Bellabeat can make an informed decision on how to encourage user engagement with their own app features.

Possible explanations could include user education on how using each feature can help them reach their fitness goal as well as effort required to use the feature creating a barrier to entry for many users. Clarifying the benefits of using various app features while making them easier to use could increase user engagement, but further research is required to know for sure.

The results of this survey show that current engagement is high for the calorie and sleep logging features and low for the activity and weight logging features. Investing resources into having quality calorie and sleep logging features in Bellabeat's own app would likely return good results. The low engagement with the activity and weight logging features could suggest that either there is untapped potential with those features that could return good results if improved, or the feature isn't desired by the users and could be removed to save development resources. More research is needed to determine the cause of the low engagement before a direction can be advised.