Dealing With Missing Data: Takeaways

Syntax

OMITTING MISSING VALUES FROM A CALCULATION

- Calculate the mean of a vector without including its missing values:

```
mean(avg_sat_score, na.rm = TRUE)
```

- Exclude from analysis any observation for which a specific variable has a missing value:

```
summary <- combined %>%
  drop_na(boro)
```

- Exclude from analysis any observation for which any variable has a missing value:

```
summary <- combined %>%
  drop_na()
```

QUANTIFYING MISSING VALUES FOR AN ENTIRE DATAFRAME

- Calculate the number of missing values for each variable in a dataframe:

```
colSums(is.na(data_frame))
```

IMPUTING TO REPLACE MISSING VALUES OF A VARIABLE

- Impute to replace missing values of a variable with a new value:

```
new_variable = replace_na(old_variable, 0))
```

Concepts

- When a data point is missing, it means no value is present for an observation of a variable. In R, missing values are represented by `NA`, which stands for "not available."

- In other words, if a variable contains any `NA` values, any summary calculations performed on that variable will result in an answer of `NA`.
- Using complete cases refers to excluding all observations for which any value in a dataframe have a missing value from your analysis.
    - Using complete cases makes sense when you're working with a dataset in which a missing value for any variable indicates that an observation should be discarded.
    - Usually, using complete cases is not advised as it can have unintended consequences for your analysis.
- Replacing missing values with appropriate substitute values is called "imputing."
- The way you choose to work with missing values in your data affects the results of your analysis, so be sure to make an informed decision by consulting metadata or the creator of the dataset.

Resources

- Documentation for na_omit()
- Documentation for replace_na()