

Handling Missing Data in the Data Science Lifecycle

Lecture Complementary Resources

https://github.com/HatefDastour/DSA_Lecture

Content:

- Lecture Slides
- Lecture Examples
- Lecture Activity



Introduction

Opening Example

Cooking and Missing Ingredients

- Imagine you're cooking a recipe but find that you're **missing key ingredients**.
- Just like missing ingredients **can ruin a dish**, missing data can lead to inaccurate analysis and poor decision-making.



* Image generated by Microsoft Designer.

The Challenge of Missing Data

- Data that is missing or incomplete.

What is Missing Data?



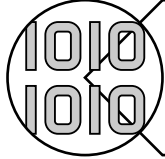
Can You Provide Examples of Missing Data?

- **Reduced Reliability:** Affects the accuracy of analysis.
- **Biased Conclusions:** Can lead to incorrect insights.
- **Model Limitations:** Many machine learning models require complete data.

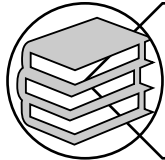
Why is it a Problem?



Goals of This Lecture



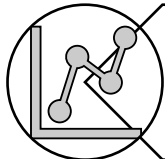
Understand Missing Data



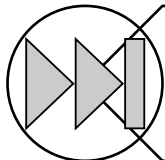
Learn Handling Techniques



Develop Critical Thinking



Practical Application



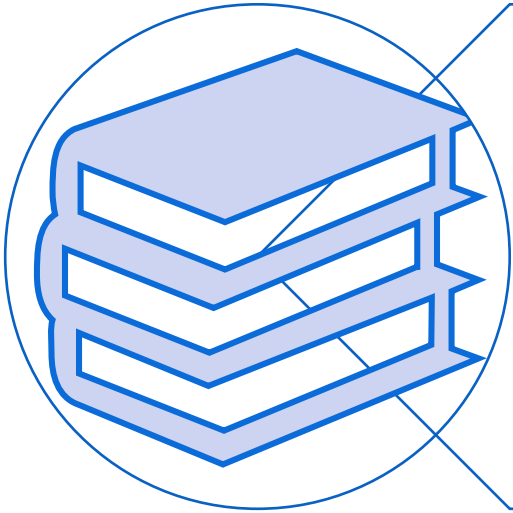
Best Practices

Types of Missing Data

Types of Missing Data

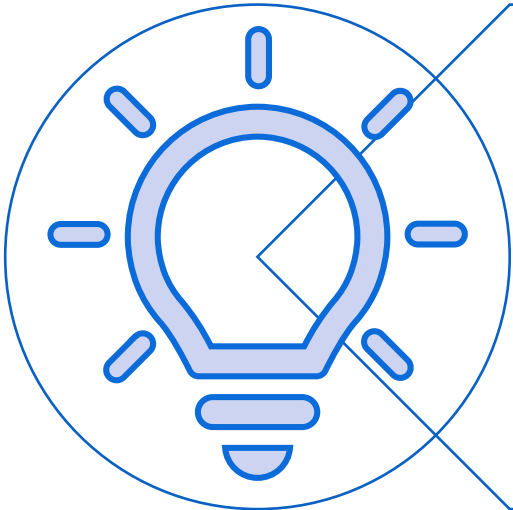


MCAR: Missing Completely at Random



Definition:

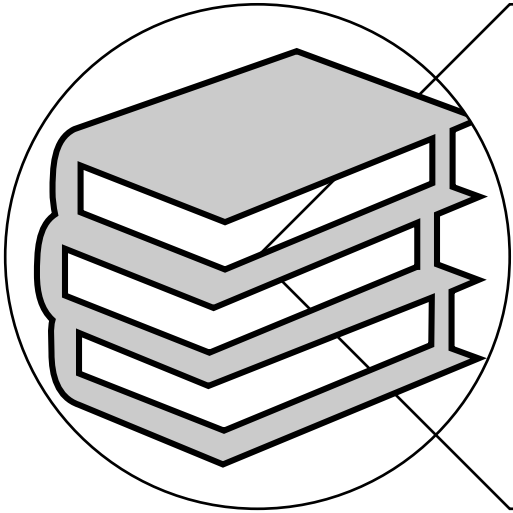
- Data missing **purely by chance**, unrelated to any characteristics or values in the dataset.



Examples:

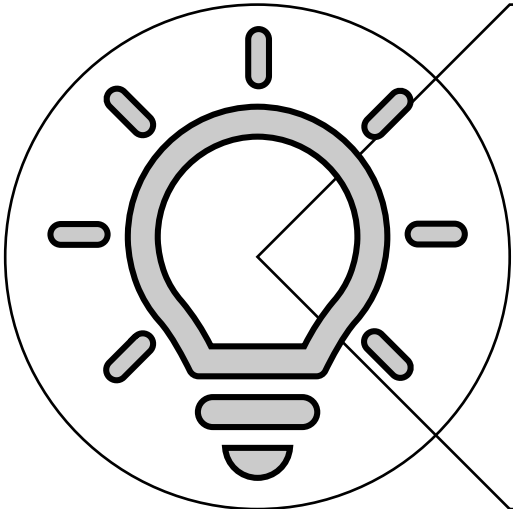
1. A shipment of completed surveys gets lost in transit.
2. A computer software randomly deletes some entries from a database due to a glitch

MAR: Missing at Random



Definition:

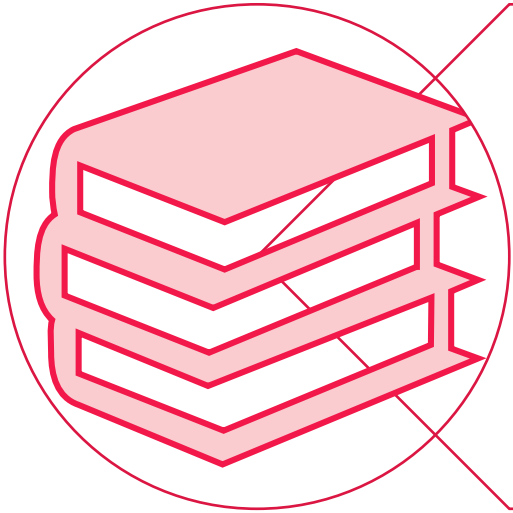
- Missingness **related to observed data**, but not to the missing data itself



Examples:

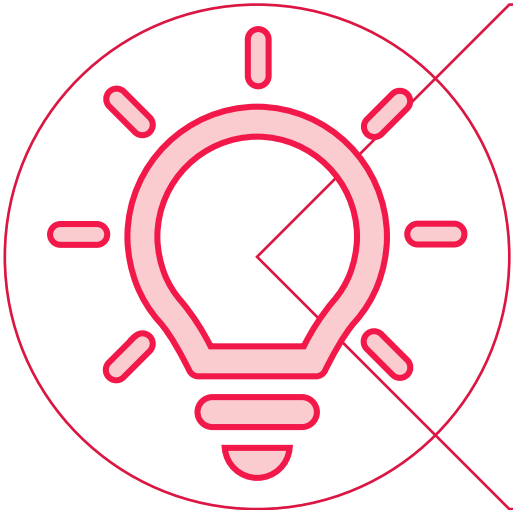
1. In a public transportation survey, participants who own cars (observed) might be more likely to skip questions about bus route preferences.
2. Participants who indicate they work night shifts (observed) might be less likely to respond to questions about daytime leisure activities.

MNAR: Missing Not at Random



Definition:

- Missingness directly related to the **unobserved data**



Examples:

1. People with low incomes (unobserved) are less likely to report their salary in a survey.
2. People who rarely read books (unobserved) might skip questions about their favorite genres in a reading habits survey.

Quiz: Types of Missing Data

1. In a medical study on a new drug, researchers record patients' initial symptom severity (mild, moderate, severe). They notice that patients initially categorized with severe symptoms are more likely to miss follow-up appointments. This is:

A) MCAR

B) MAR

C) MNAR

2. In a workplace survey, employees with lower job satisfaction are less likely to respond to questions about their work environment. However, the researchers do not know the actual satisfaction levels of those who didn't respond. This represents:

A) MCAR

B) MAR

C) MNAR

3. A researcher accidentally deletes a random selection of data entries while cleaning up a dataset. This scenario is:

A) MCAR

B) MAR

C) MNAR

Identifying Missing Data



Requirements for the Next Part

1. Have Basic Statistical Knowledge:

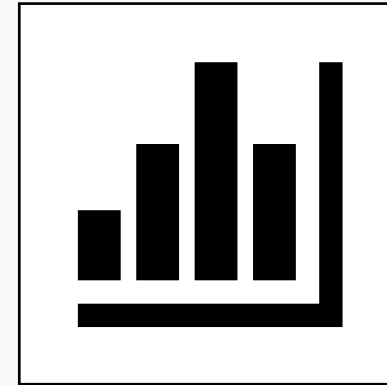
- Familiar with concepts such as *mean and median*.

2. Understand Python Basics:

- Basic understanding of *Python* and libraries such as *NumPy* and *Pandas*.

3. Access to Google Colab:

- Have access to *Google Colab* and know the basics of *Jupyter Notebook*.



Lecture Examples

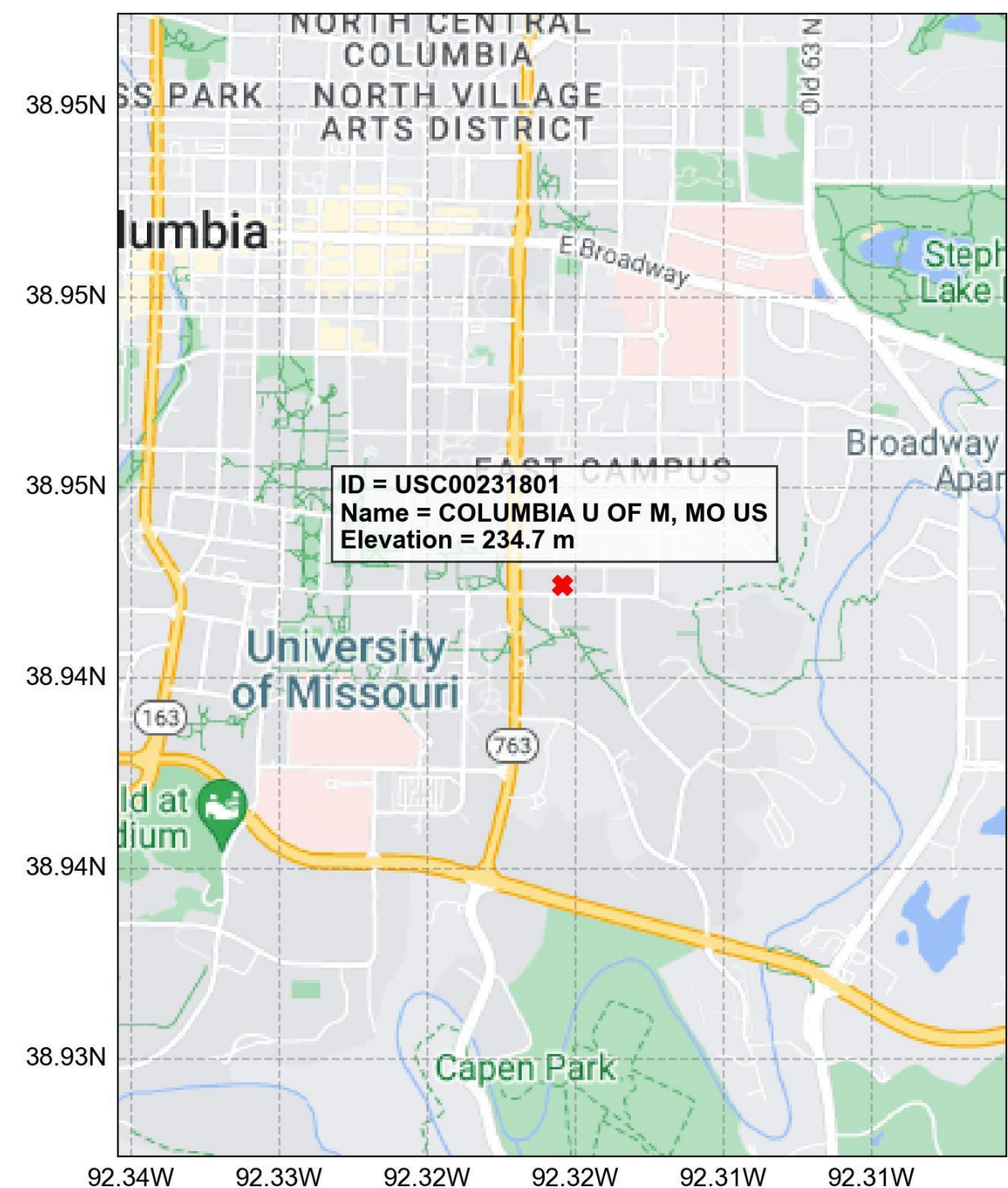
https://github.com/HatefDastour/DSA_Lecture

Lecture_Examples.ipynb



Weather Data Example (1/3)

- This dataset contains daily temperature readings for Columbia, Missouri, specifically from the **University of Missouri weather station**
- **Data Source:** [NCEI Climate Data](#)
- **Period:** October 01, 2024, to October 10, 2024
- **Note:** Some data points have been intentionally removed to create a time series with missing values for educational purposes.



Weather Data Example (2/3)

https://raw.githubusercontent.com/HatefDastour/DSA_Lecture/refs/heads/main/data_files/data_10day_standard_missing.csv

	TMIN	TMAX	
Minimum Daily Temperature (°F)	2024-10-01	49.0	72.0
	2024-10-02	44.0	74.0
	2024-10-03	NaN	NaN
	2024-10-04	NaN	83.0
	2024-10-05	62.0	89.0
	2024-10-06	NaN	79.0
	2024-10-07	47.0	72.0
	2024-10-08	NaN	NaN
	2024-10-09	46.0	78.0
	2024-10-10	49.0	80.0
	10 rows × 2 columns		Maximum Daily Temperature (°F)

Weather Data Example (3/3)

	TMIN	TMAX
2024-10-01	49.0	72.0
2024-10-02	44.0	74.0
2024-10-03	NaN	NaN
2024-10-04	NaN	83.0
2024-10-05	62.0	89.0
2024-10-06	NaN	79.0
2024-10-07	47.0	72.0
2024-10-08	NaN	NaN
2024-10-09	46.0	78.0
2024-10-10	49.0	80.0

10 rows × 2 columns



	TMIN	TMAX
2024-10-01	False	False
2024-10-02	False	False
2024-10-03	True	True
2024-10-04	True	False
2024-10-05	False	False
2024-10-06	True	False
2024-10-07	False	False
2024-10-08	True	True
2024-10-09	False	False
2024-10-10	False	False

10 rows × 2 columns

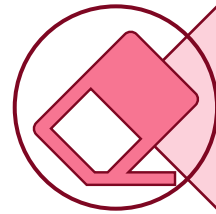
Utilizing Pandas' *isna()* and *isnull()* functions for missing data detection.

Common Methods for Handling Missing Data



Approaches to Handling Missing Values

When it comes to missing values in datasets, we can take two primary approaches:



Dropping Missing Values



Imputing Missing Values

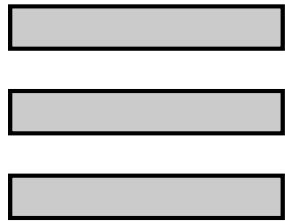
Note: In order for these methods to produce appropriate results in most situations, data must be what is known as Missing Completely At Random (**MCAR**).

Dropping Missing Values



Listwise Deletion

- Removes **any row** with missing data.



Pairwise Deletion

- Uses all available data for each analysis, excluding only the missing values for that specific analysis.

Listwise Deletion: Example

	TMIN	TMAX
2024-10-01	49.0	72.0
2024-10-02	44.0	74.0
2024-10-03	NaN	NaN
2024-10-04	NaN	83.0
2024-10-05	62.0	89.0
2024-10-06	NaN	79.0
2024-10-07	47.0	72.0
2024-10-08	NaN	NaN
2024-10-09	46.0	78.0
2024-10-10	49.0	80.0

10 rows × 2 columns

`dropna(how = 'any')`

What are the
differences?

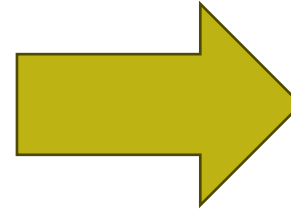
	TMIN	TMAX
2024-10-01	49.0	72.0
2024-10-02	44.0	74.0
2024-10-05	62.0	89.0
2024-10-07	47.0	72.0
2024-10-09	46.0	78.0
2024-10-10	49.0	80.0

6 rows × 2 columns

Pairwise Deletion: Example

	TMIN	TMAX
2024-10-01	49.0	72.0
2024-10-02	44.0	74.0
2024-10-03	NaN	NaN
2024-10-04	NaN	83.0
2024-10-05	62.0	89.0
2024-10-06	NaN	79.0
2024-10-07	47.0	72.0
2024-10-08	NaN	NaN
2024-10-09	46.0	78.0
2024-10-10	49.0	80.0

10 rows × 2 columns



Excludes NaN values

```
climate_data[ 'TMIN' ].mean(skipna=True)  
climate_data[ 'TMAX' ].mean(skipna=True)
```

Using Pairwise Deletion:

- Mean TMIN (using pairwise deletion): 49.50 °F
- Mean TMAX (using pairwise deletion): 78.38 °F

Using Listwise Deletion:

- Mean TMIN (after listwise deletion): 49.50 °F
- Mean TMAX (after listwise deletion): 77.50 °F

Dropping Missing Values: Pros and Cons

- **Pros:** Simple, preserves potential relationships.
- **Cons:** Reduces sample size, potential for bias.

Listwise Deletion



- **Pros:** Retains more data.
- **Cons:** Inconsistent sample sizes, potential for bias.

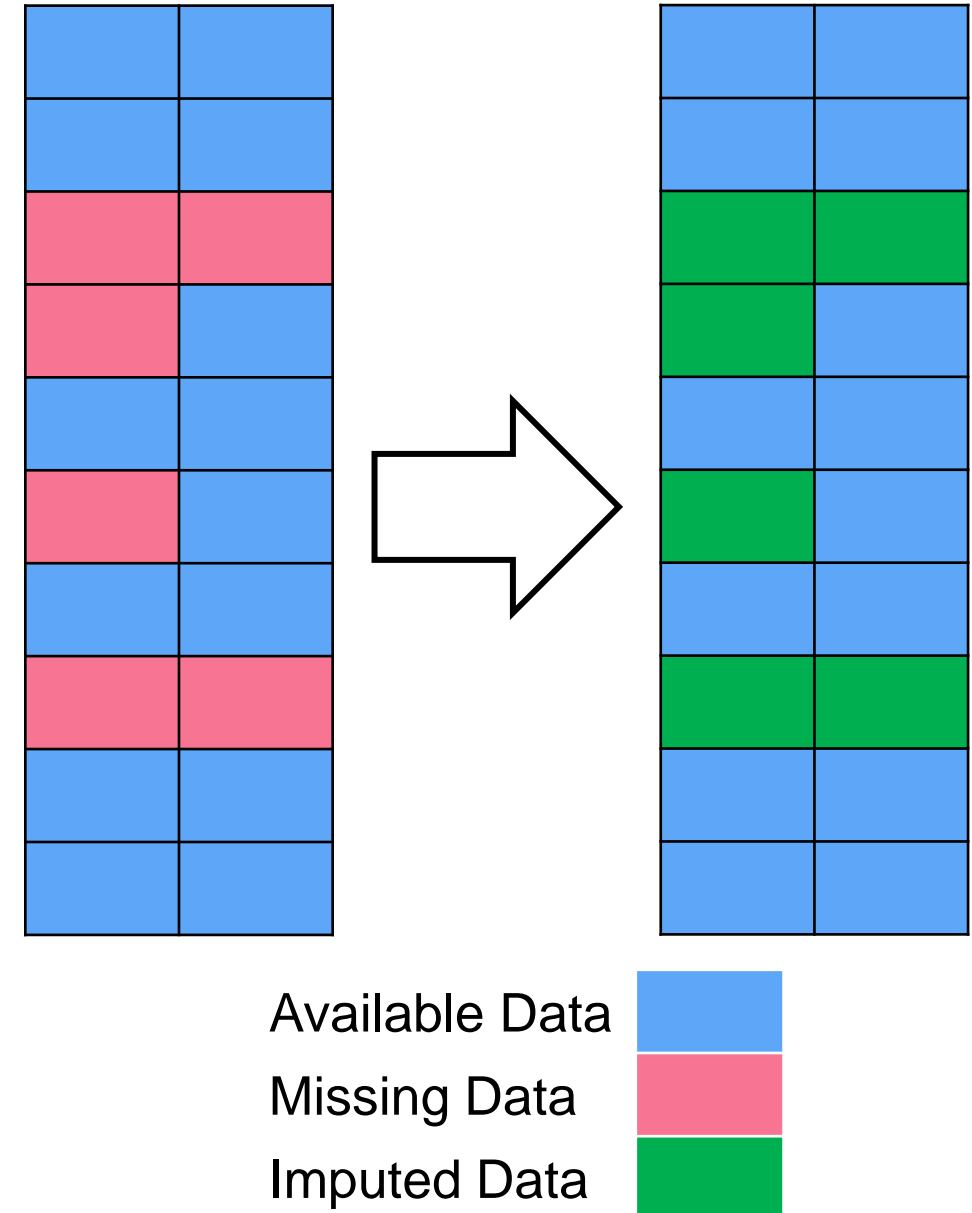
Pairwise Deletion



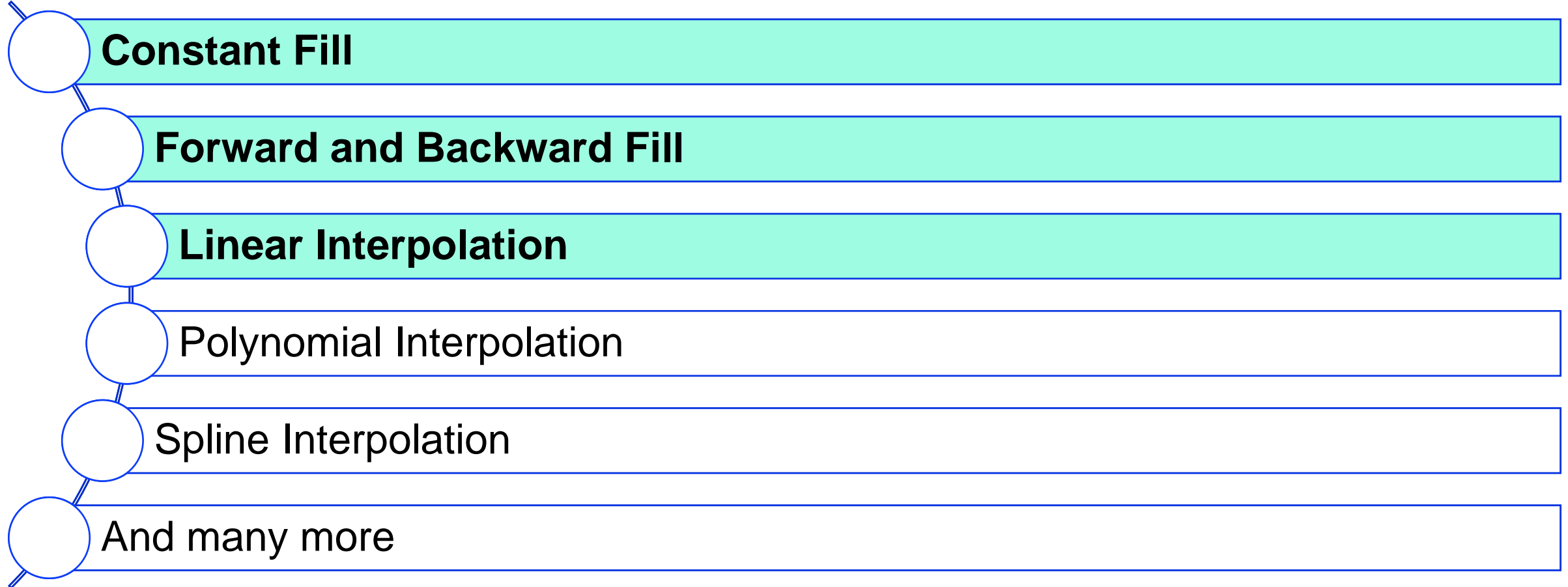
Imputing Missing Values

- Imputation involves **filling in missing values** using various techniques.
- This approach helps to **maintain sample size** and can improve the accuracy of analyses.

Conceptual Figure



Imputation Methods



Constant Fill

Definition: Replaces missing values with a **specified constant value** (e.g., zero, mean, median, or another meaningful number).

	TMIN
2024-10-01	49.0
2024-10-02	44.0
2024-10-03	NaN
2024-10-04	NaN
2024-10-05	62.0
2024-10-06	NaN
2024-10-07	47.0
2024-10-08	NaN
2024-10-09	46.0
2024-10-10	49.0

Constant Fill with a
Constant Value

	TMIN
2024-10-01	49.0
2024-10-02	44.0
2024-10-03	Constant Value
2024-10-04	Constant Value
2024-10-05	62.0
2024-10-06	Constant Value
2024-10-07	47.0
2024-10-08	Constant Value
2024-10-09	46.0
2024-10-10	49.0

Constant Fill – Example (1/2)

	TMIN
2024-10-01	49.0
2024-10-02	44.0
2024-10-03	NaN
2024-10-04	NaN
2024-10-05	62.0
2024-10-06	NaN
2024-10-07	47.0
2024-10-08	NaN
2024-10-09	46.0
2024-10-10	49.0

Constant Fill with
Mean

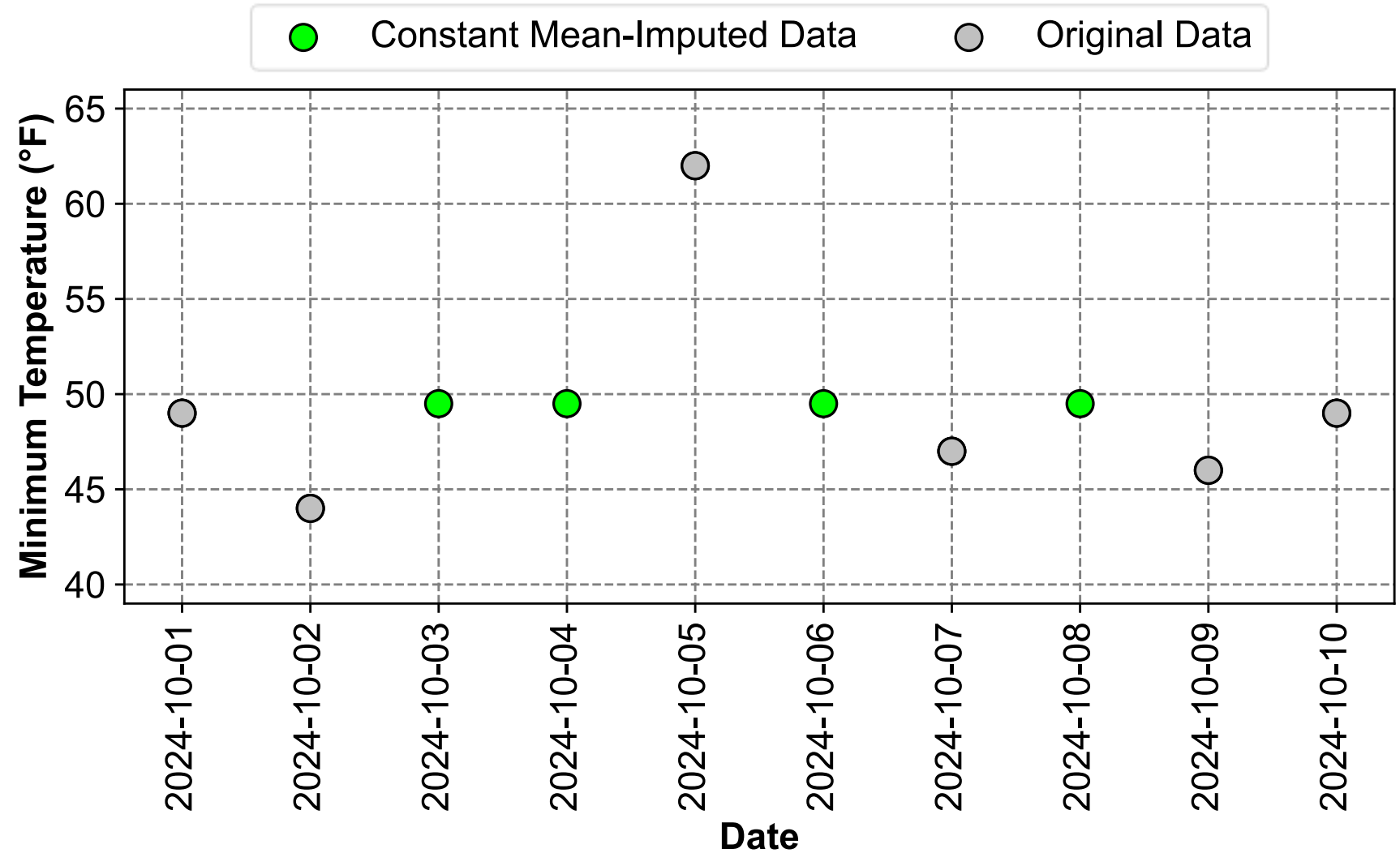
Mean of TMIN = 49.50 °F

	TMIN
2024-10-01	49.0
2024-10-02	44.0
2024-10-03	49.5
2024-10-04	49.5
2024-10-05	62.0
2024-10-06	49.5
2024-10-07	47.0
2024-10-08	49.5
2024-10-09	46.0
2024-10-10	49.0

Constant Fill – Example (2/2)

	TMIN
2024-10-01	49.0
2024-10-02	44.0
2024-10-03	49.5
2024-10-04	49.5
2024-10-05	62.0
2024-10-06	49.5
2024-10-07	47.0
2024-10-08	49.5
2024-10-09	46.0
2024-10-10	49.0

Mean of TMIN = 49.50 °F



Constant Fill – Benefits and Considerations

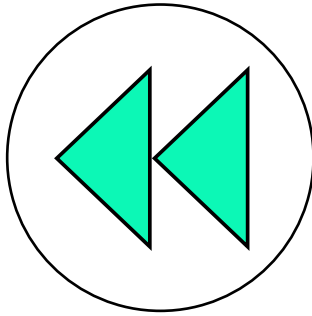
Benefits

- **Simplicity:** Easy to implement and understand
- **Contextual Relevance:** Effective when a default value is logical (e.g., zero for missing income)

Considerations

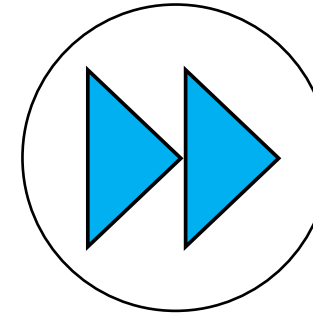
- **Potential Bias:** May not accurately represent the true data distribution
- **Reduced Variability:** Can affect statistical analyses and model performance

Backward Fill and Forward Fill



Backward Fill (bfill):

Fills missing values using the **next valid observation**



Forward Fill (ffill):

Fills missing values using the **last valid observation**



Backward Fill: Example (1/2)

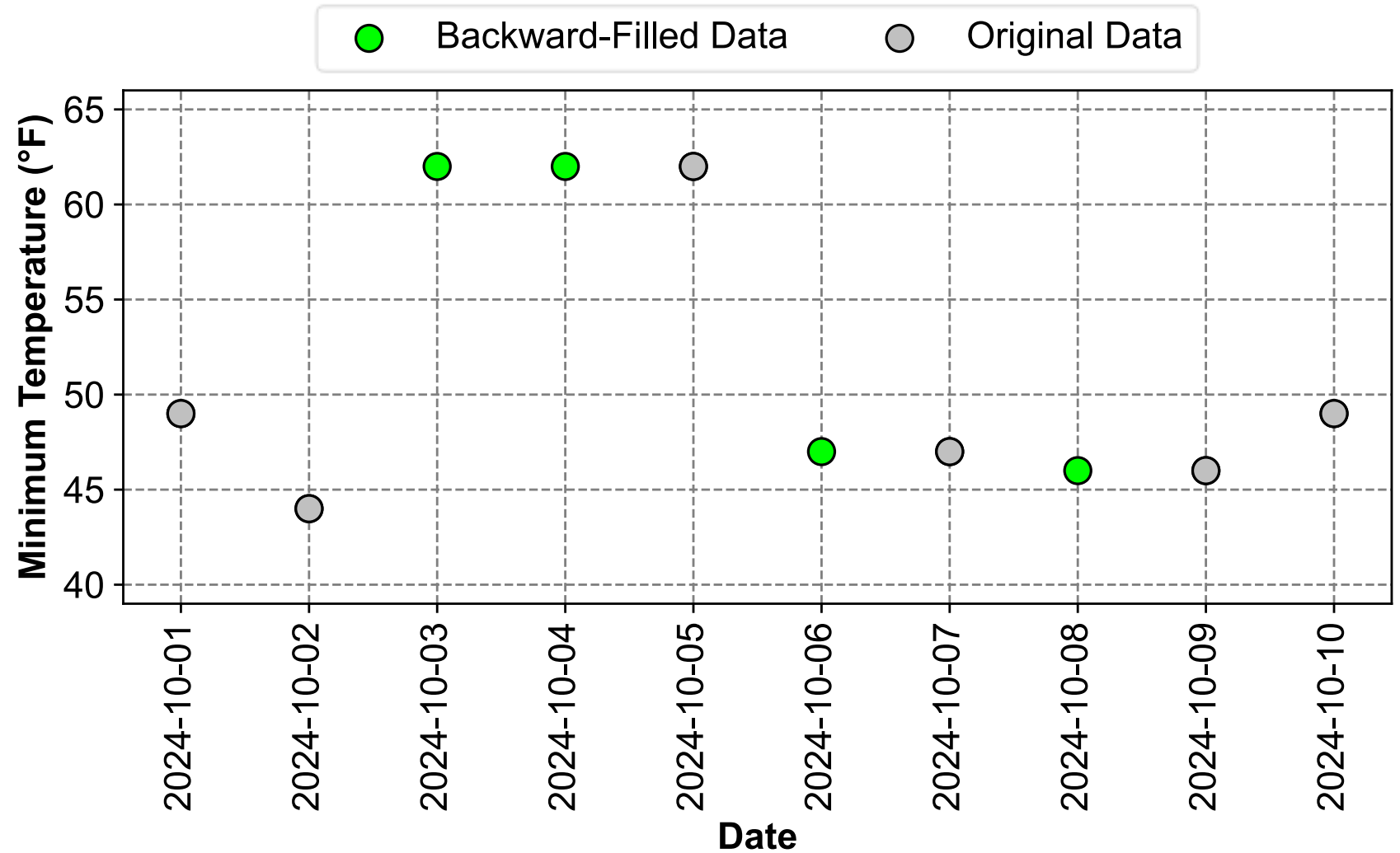
	TMIN
2024-10-01	49.0
2024-10-02	44.0
2024-10-03	NaN
2024-10-04	NaN
2024-10-05	62.0
2024-10-06	NaN
2024-10-07	47.0
2024-10-08	NaN
2024-10-09	46.0
2024-10-10	49.0

Backward Fill

	TMIN
2024-10-01	49.0
2024-10-02	44.0
2024-10-03	62.0
2024-10-04	62.0
2024-10-05	62.0
2024-10-06	47.0
2024-10-07	47.0
2024-10-08	46.0
2024-10-09	46.0
2024-10-10	49.0

Backward Fill: Example (2/2)

	TMIN
2024-10-01	49.0
2024-10-02	44.0
2024-10-03	62.0
2024-10-04	62.0
2024-10-05	62.0
2024-10-06	47.0
2024-10-07	47.0
2024-10-08	46.0
2024-10-09	46.0
2024-10-10	49.0



Forward Fill: Example (1/2)

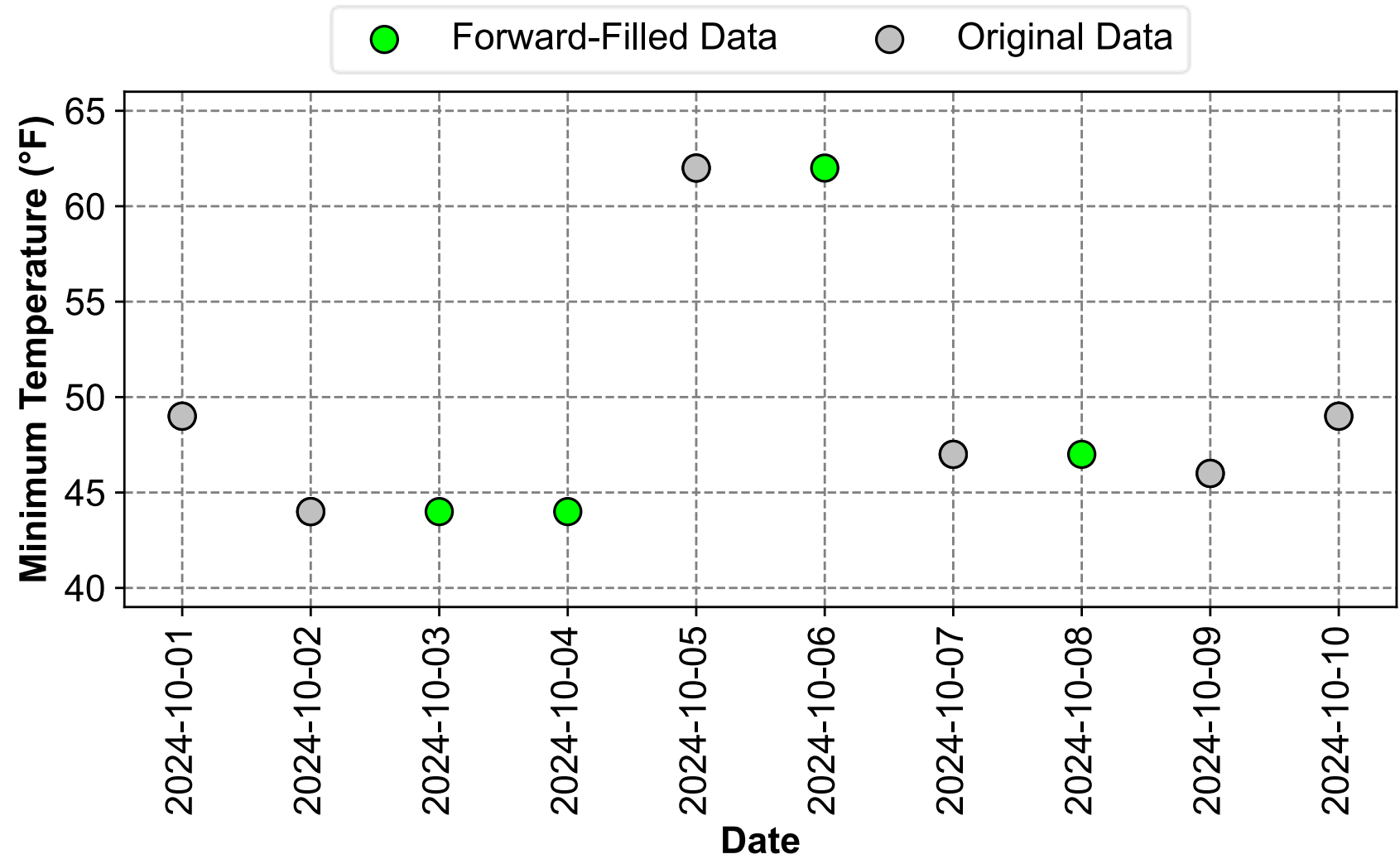
	TMIN
2024-10-01	49.0
2024-10-02	44.0
2024-10-03	NaN
2024-10-04	NaN
2024-10-05	62.0
2024-10-06	NaN
2024-10-07	47.0
2024-10-08	NaN
2024-10-09	46.0
2024-10-10	49.0

Forward Fill

	TMIN
2024-10-01	49.0
2024-10-02	44.0
2024-10-03	44.0
2024-10-04	44.0
2024-10-05	62.0
2024-10-06	62.0
2024-10-07	47.0
2024-10-08	47.0
2024-10-09	46.0
2024-10-10	49.0

Forward Fill: Example (2/2)

	TMIN
2024-10-01	49.0
2024-10-02	44.0
2024-10-03	44.0
2024-10-04	44.0
2024-10-05	62.0
2024-10-06	62.0
2024-10-07	47.0
2024-10-08	47.0
2024-10-09	46.0
2024-10-10	49.0



Backward Fill and Forward Fill: Benefits and Considerations

Benefits

- **Simplicity:** Easy to implement and understand
- **Time Series Relevance:** Particularly useful for time-based data
- **Preserves Trends:** Maintains data patterns within a series

Considerations

- **Accuracy Limitations:** May not reflect true values, especially for long gaps
- **Directional Bias:** Forward fill favors past data; backward fill favors future data
- **Data Dependency:** Effectiveness relies on the nature and frequency of available data points

Linear Interpolation

What is Linear Interpolation?

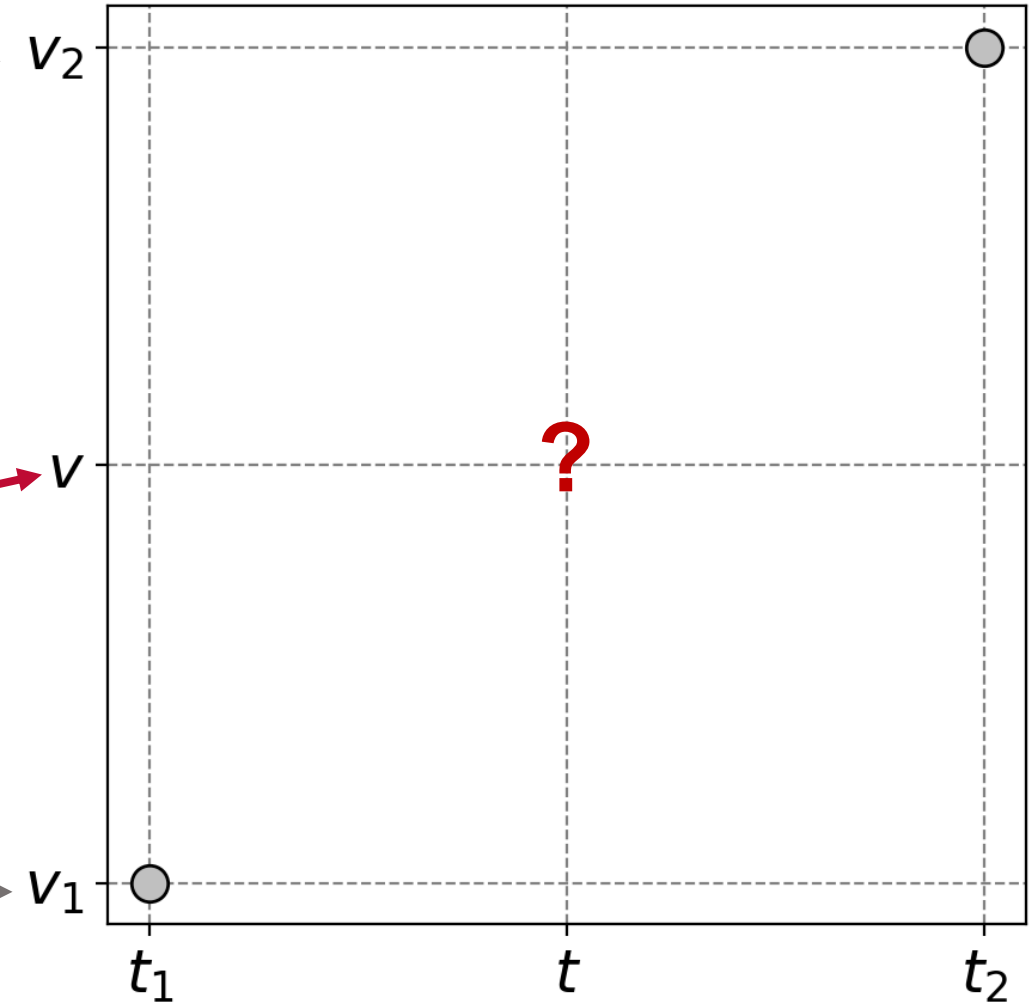
- Linear interpolation estimates missing values in time series data by assuming a straight line between known data points.

Linear Interpolation: Explanation (1/3)

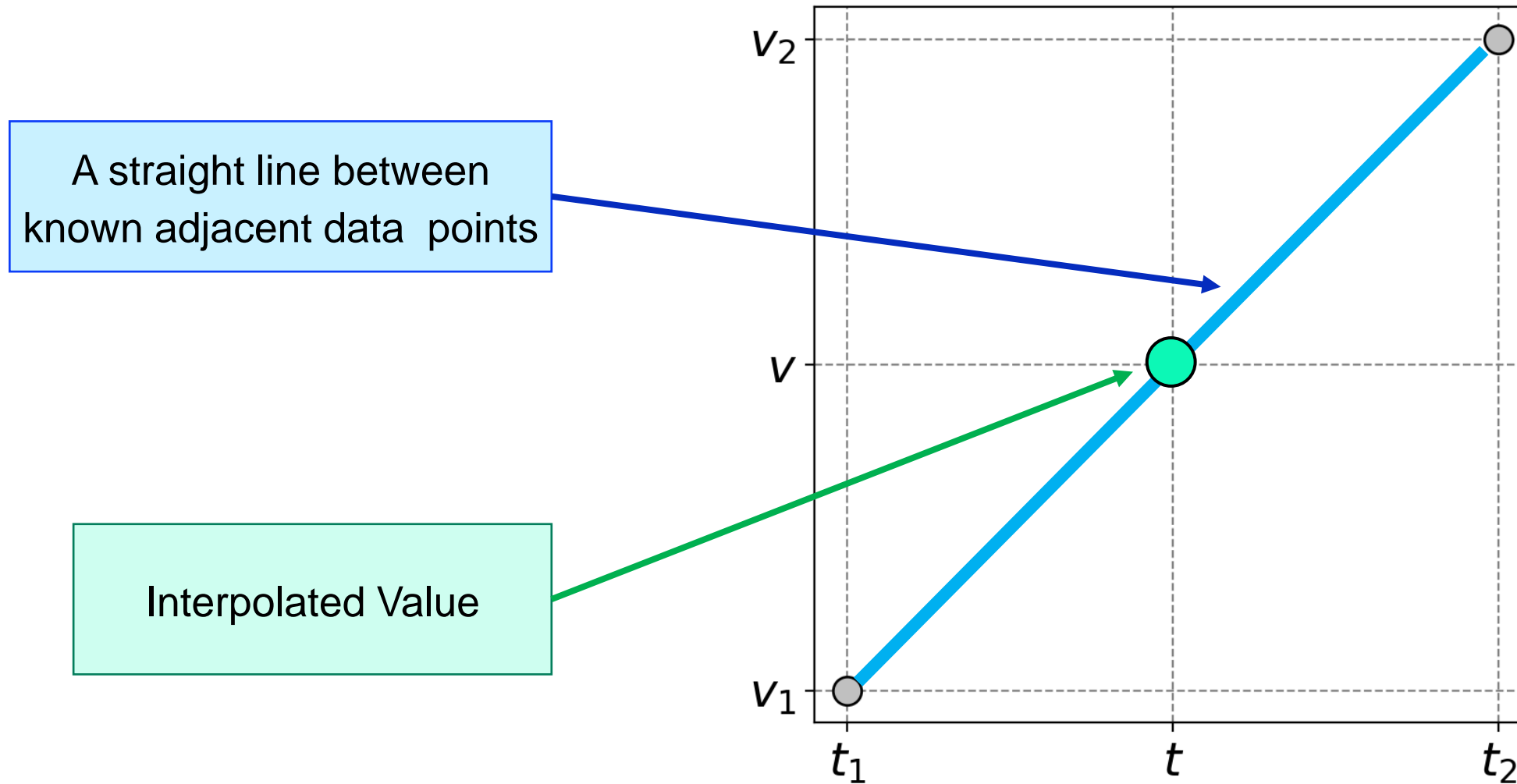
- t_2 is the time of the known point.
- v_2 is the y-value at time t_2

- t is the time for which we want to find the v-value.
- v is the missing value at time t

- t_1 is the time of the known point.
- v_1 is the y-value at time t_1 .



Linear Interpolation: Explanation (2/3)



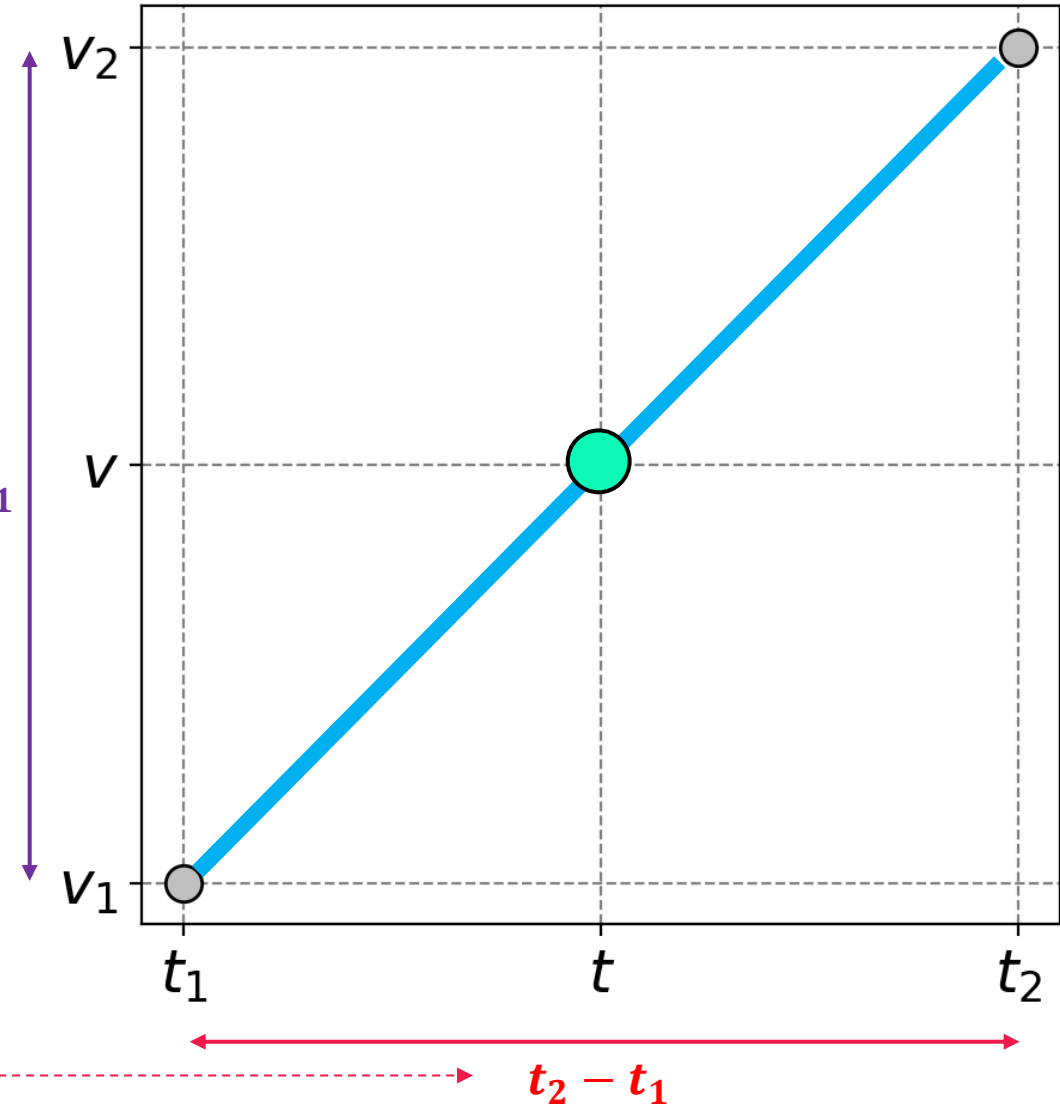
Linear Interpolation: Explanation (3/3)

- m is the Slope with

$$m = \frac{(v_2 - v_1)}{(t_2 - t_1)}$$

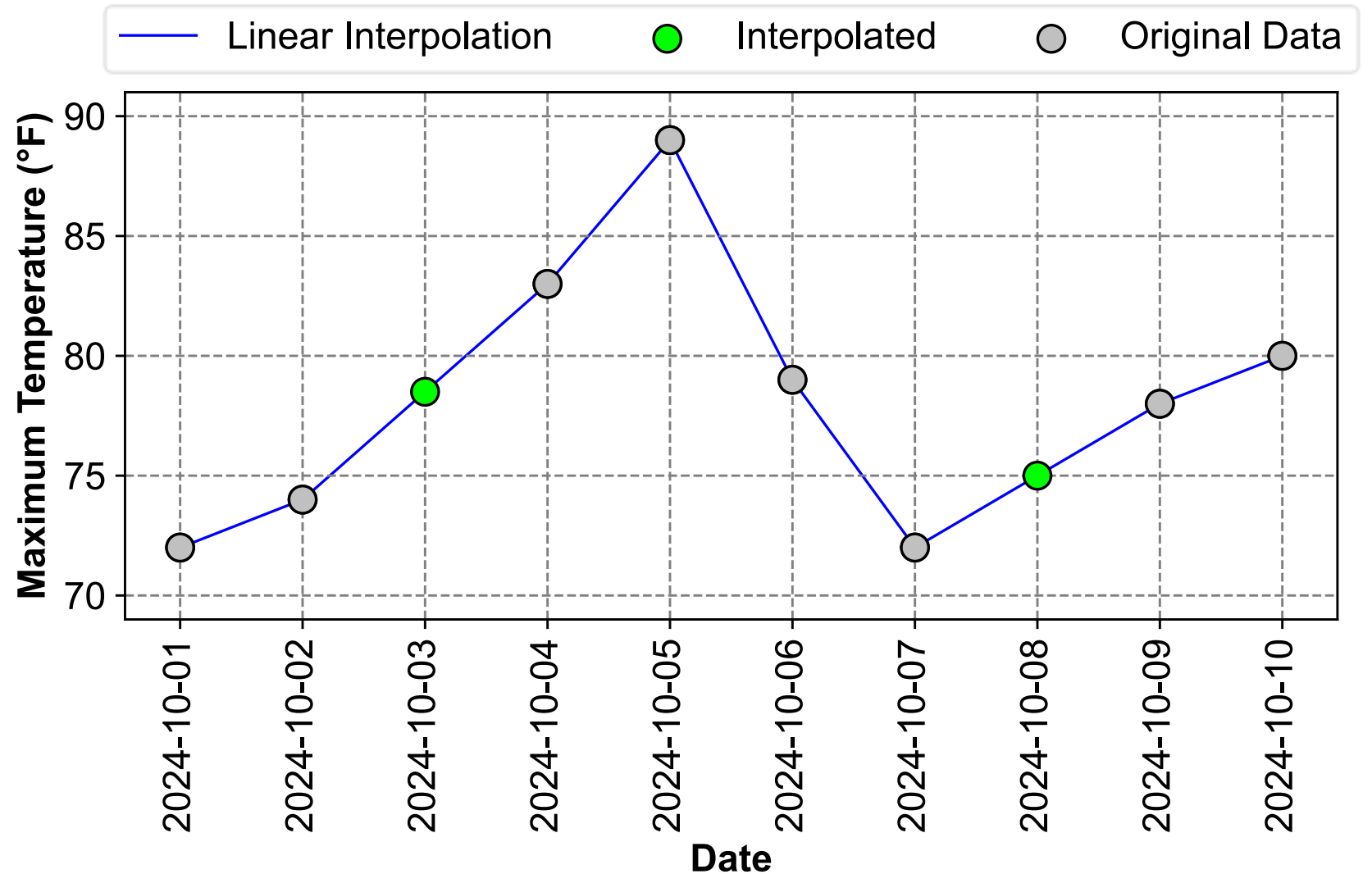
- Linear Interpolation:

$$v = v_1 + m * (t - t_1)$$



Linear Interpolation: Example

	TMAX
2024-10-01	72.0
2024-10-02	74.0
2024-10-03	78.5
2024-10-04	83.0
2024-10-05	89.0
2024-10-06	79.0
2024-10-07	72.0
2024-10-08	75.0
2024-10-09	78.0
2024-10-10	80.0



Linear Interpolation: Benefits and Considerations

Benefits

- **Simplicity:** Easy to understand and implement
- **Computational Efficiency:** Fast to calculate, even for large datasets
- **Predictability:** Results are consistent and easily reproducible

Considerations

- **Accuracy Limitations:** May not capture complex, non-linear relationships
- **Curve Smoothness:** Can result in sharp transitions between data points
- **Boundary Issues:** Doesn't work for missing values at the edges of the dataset.

Lecture Activity

https://github.com/HatefDastour/DSA_Lecture

Lecture_Activity.ipynb



Open in Colab



Thank You

Hatef Dastour

✉ hatef.dastour@ucalgary.ca

