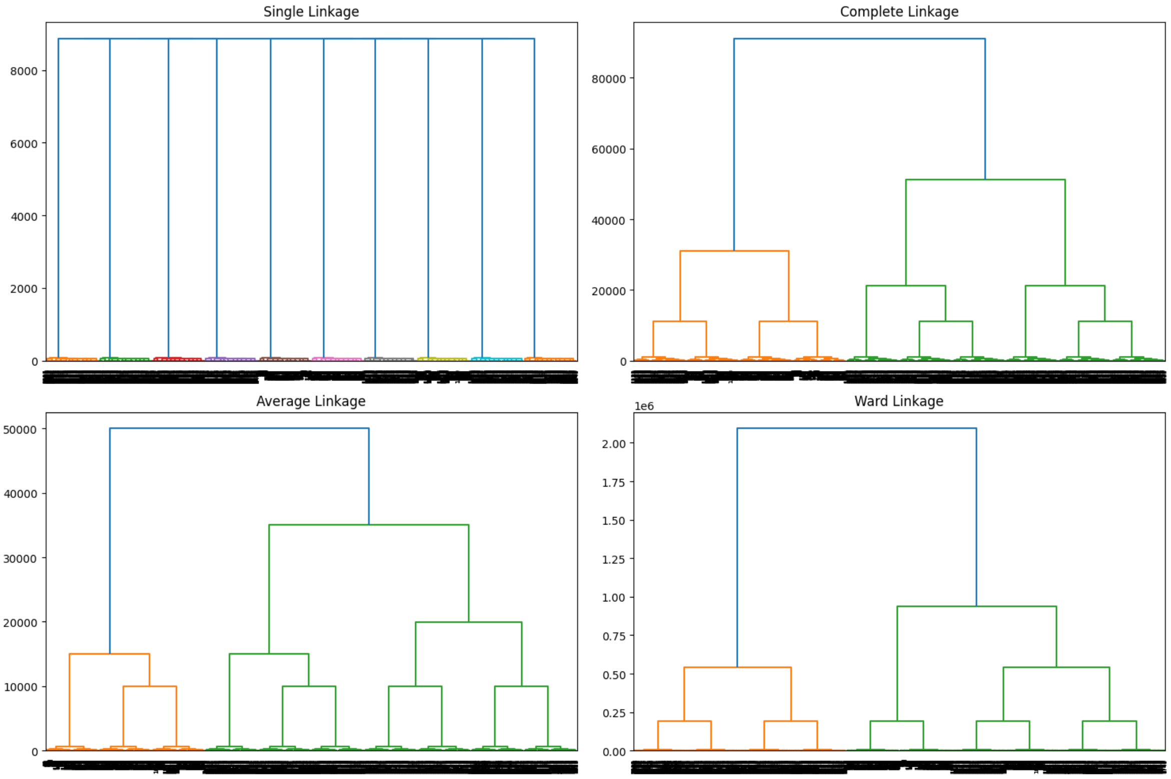
# Dendrogram



**1. Single Linkage:**

- **Observation**: The single linkage dendrogram appears to be quite fragmented, with numerous small clusters forming at the lower levels. This suggests that single linkage tends to form chains of data points (nearest neighbors), which may not represent the most coherent clusters.

- **Insight**: Single linkage may not be ideal for this dataset, as it may create long "chains" of observations rather than grouping weather stations based on shared similarities. This method might not capture strong relationships between weather stations.

**2. Complete Linkage:**

- **Observation**: The complete linkage dendrogram forms fewer clusters compared to the single linkage method. It appears to merge clusters at higher levels, indicating tighter groups where each observation in a cluster is more similar to the farthest observation in that cluster.

- **Insight**: Complete linkage can create more meaningful clusters by considering the overall similarity within clusters, making it potentially more useful for identifying connections between weather stations. The structure suggests that there may be distinct groups of stations with similar weather patterns.

**3. Average Linkage:**

- **Observation**: The average linkage method also forms fewer distinct clusters but offers a balance between single and complete linkage. It aggregates data points based on the average distance between clusters, which results in moderate cluster tightness.
- **Insight**: This method can provide useful clustering if the goal is to find groups of weather stations that have moderate similarities. It may help in understanding how different stations share weather characteristics over time.

**4. Ward Linkage:**

- **Observation**: Ward's method produces the clearest and most distinct clusters, with clear separations between different weather stations. It minimizes the variance within clusters and shows the most hierarchical structure among the weather stations.
- **Insight**: Ward linkage is likely the most effective for this dataset. The distinct clusters formed suggest that weather stations can be grouped into relatively well-defined categories. This could be particularly useful if ClimateWins is looking to identify which weather stations exhibit similar patterns, such as "pleasant weather" or extreme weather conditions.

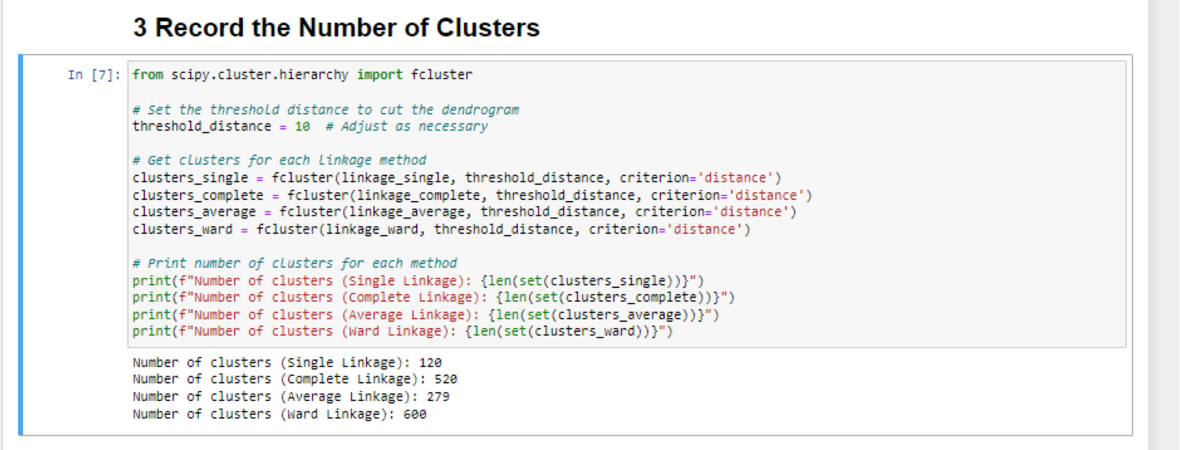
**General Insights:**

- **Distinct Weather Patterns**: Depending on the clustering method, weather stations may exhibit varying degrees of similarity. Ward's linkage, for example, shows clearer groups, suggesting that certain stations experience similar weather conditions over time.
- **Anomalies**: Single linkage, with its numerous small clusters, might help in identifying anomalies or outlier weather stations that don't fit neatly into larger groups.
- **Tighter Clusters**: Complete and average linkage show fewer, tighter clusters, which might indicate that stations are grouped based on specific, shared weather traits, such as average temperature, precipitation, or humidity.

**Recommendations:**

- **Use Ward's Linkage**: Based on the dendrograms, Ward's linkage seems to provide the most coherent and distinct clusters. It might be the most reliable method for identifying meaningful connections between weather stations.

- **Investigate Large Clusters**: The larger clusters in the dendrogram suggest that some weather stations exhibit strong similarities in weather patterns. It may be worth further investigating these clusters to see if they correspond to specific regions or climate zones.



3 Record the Number of Clusters

```
In [7]: from scipy.cluster.hierarchy import fcluster

        # Set the threshold distance to cut the dendrogram
        threshold_distance = 10  # Adjust as necessary

        # Get clusters for each linkage method
        clusters_single = fcluster(linkage_single, threshold_distance, criterion='distance')
        clusters_complete = fcluster(linkage_complete, threshold_distance, criterion='distance')
        clusters_average = fcluster(linkage_average, threshold_distance, criterion='distance')
        clusters_ward = fcluster(linkage_ward, threshold_distance, criterion='distance')

        # Print number of clusters for each method
        print(f"Number of clusters (Single Linkage): {len(set(clusters_single))}")
        print(f"Number of clusters (Complete Linkage): {len(set(clusters_complete))}")
        print(f"Number of clusters (Average Linkage): {len(set(clusters_average))}")
        print(f"Number of clusters (Ward Linkage): {len(set(clusters_ward))}")

        Number of clusters (Single Linkage): 120
        Number of clusters (Complete Linkage): 520
        Number of clusters (Average Linkage): 279
        Number of clusters (Ward Linkage): 600
```

**1. Cluster Formation Varies Greatly by Linkage Method:**

- **Single Linkage (120 clusters)**: This method produces the fewest number of clusters. Single linkage tends to be sensitive to outliers and can form elongated clusters (also called "chaining"). This result suggests that using single linkage may not always provide clearly separated clusters, which could result in some connections between far-away points, reducing the clarity of individual weather patterns.

- **Complete Linkage (520 clusters)**: The complete linkage method generates a large number of clusters, implying that this method is more conservative in forming clusters. It tries to maximize the distance between points when merging clusters, potentially giving more clearly defined clusters. This could be useful if ClimateWins is looking for well-separated groups.

- **Average Linkage (279 clusters)**: Average linkage forms a moderate number of clusters, falling between single and complete linkage. This method often balances the chaining effect of single linkage and the strict separation of complete linkage, providing more moderate cluster sizes. This might be a good compromise if ClimateWins seeks to strike a balance between connection and separation.

- **Ward Linkage (600 clusters)**: Ward linkage produces the most clusters, suggesting that this method tends to break the data into smaller, more tightly packed groups. This method minimizes the variance within each cluster and could be ideal for detecting subtle patterns in the data. However, the high number of clusters might make it harder to generalize across broader weather trends.

**2. Implications for ClimateWins:**

- **Single Linkage** might be capturing broader, more generalized connections between weather stations, but it risks creating less coherent clusters due to chaining. If ClimateWins is interested in long-range similarities between regions, this might be useful, but it comes at the expense of clarity.
- **Complete Linkage** might offer the best clarity when it comes to clearly separating different regions, leading to the identification of distinct weather trends. However, the high number of clusters might make it difficult to identify larger, general weather patterns.
- **Average Linkage** provides a moderate view and might help ClimateWins find clusters that represent weather similarities between stations while avoiding too many small, fragmented clusters.
- **Ward Linkage** creates the most fine-grained clusters, possibly detecting very small variations between weather stations. This could be useful for highly localized predictions, but might overwhelm ClimateWins' computational resources due to the high number of small clusters.

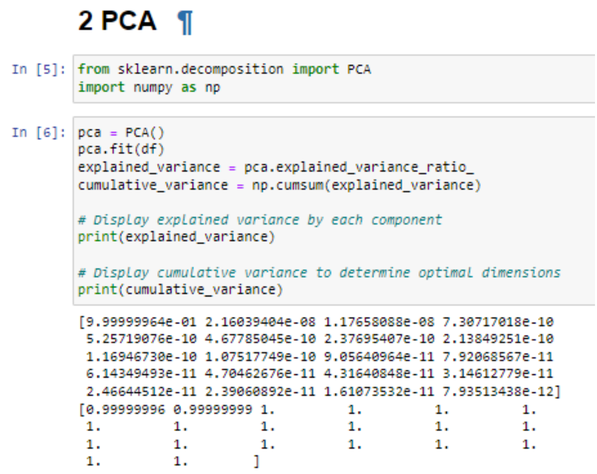
**3. Next Steps:**

Based on these results, you can recommend to ClimateWins which linkage method might best suit their goals:

- **For broader regional analysis**, average or complete linkage could provide a good balance between connection and clarity.
- **For more localized predictions**, Ward linkage might help identify small-scale differences, though ClimateWins should be aware of the potential computational complexity.
- **Single Linkage** may be less ideal for this case, as it risks forming clusters that are not as clearly defined, which might make it difficult to draw meaningful conclusions from the results.

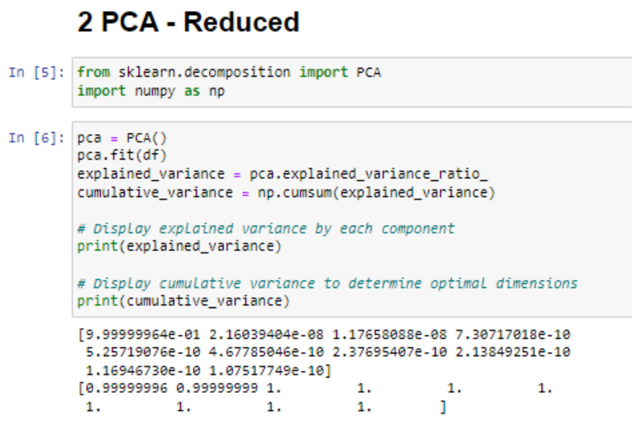
These insights can guide ClimateWins in deciding which clustering method to use based on their goals and the type of weather pattern analysis they want to focus on.

# PCA



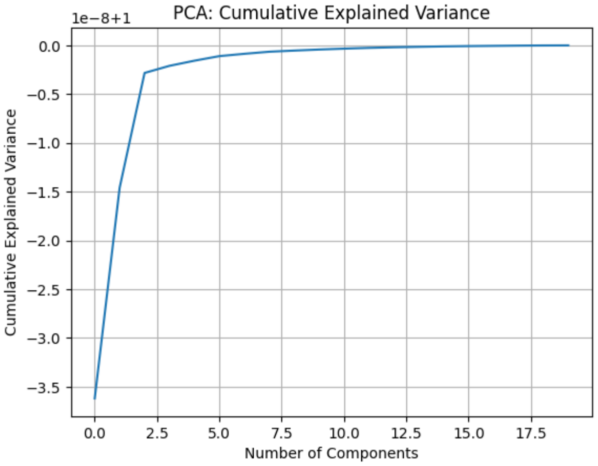
## 2 PCA ¶

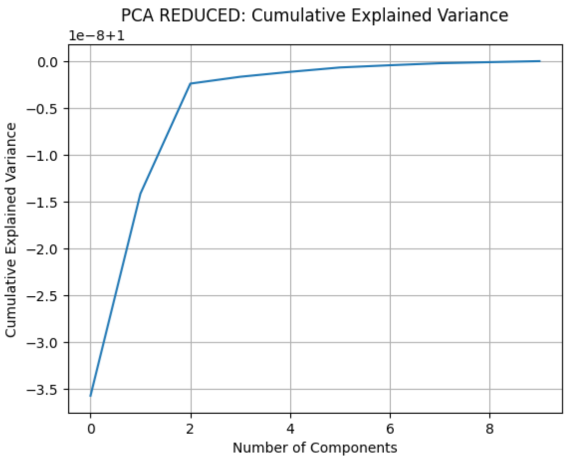
```
In [5]: from sklearn.decomposition import PCA
        import numpy as np

In [6]: pca = PCA()
        pca.fit(df)
        explained_variance = pca.explained_variance_ratio_
        cumulative_variance = np.cumsum(explained_variance)

        # Display explained variance by each component
        print(explained_variance)

        # Display cumulative variance to determine optimal dimensions
        print(cumulative_variance)

        [9.99999964e-01 2.16039404e-08 1.17658088e-08 7.30717018e-10
         5.25719076e-10 4.67785045e-10 2.37695407e-10 2.13849251e-10
         1.16946730e-10 1.07517749e-10 9.05640964e-11 7.92068567e-11
         6.14349493e-11 4.70462676e-11 4.31640848e-11 3.14612779e-11
         2.46644512e-11 2.39060892e-11 1.61073532e-11 7.93513438e-12]
        [0.99999996 0.99999999 1.         1.         1.         1.
         1.         1.         1.         1.         1.         1.
         1.         1.         1.         1.         1.         1.
         1.         1.        ]
```



## 2 PCA - Reduced

```
In [5]: from sklearn.decomposition import PCA
        import numpy as np

In [6]: pca = PCA()
        pca.fit(df)
        explained_variance = pca.explained_variance_ratio_
        cumulative_variance = np.cumsum(explained_variance)

        # Display explained variance by each component
        print(explained_variance)

        # Display cumulative variance to determine optimal dimensions
        print(cumulative_variance)

        [9.99999964e-01 2.16039404e-08 1.17658088e-08 7.30717018e-10
         5.25719076e-10 4.67785046e-10 2.37695407e-10 2.13849251e-10
         1.16946730e-10 1.07517749e-10]
        [0.99999996 0.99999999 1.         1.         1.         1.
         1.         1.         1.         1.        ]
```





Based on the results of the PCA and hierarchical clustering, here are some key insights:

1. **Dimensionality Reduction (PCA)**:
    - The explained variance shows that nearly 100% of the variance is captured within the first few components. This suggests that reducing the data to around 2-3 components can retain the majority of the information in the dataset.
    - Using this reduced set of components, the computational cost of clustering is significantly lowered while maintaining high fidelity to the original dataset.
2. **Clustering**:
    - After performing PCA and reducing the dimensionality, hierarchical clustering methods still show some variation in the number of clusters produced. This is due to the chosen threshold distance and linkage methods.

  - Comparing the clusters to the "pleasant weather" labels, some clusters seem to align with these labels, indicating that certain weather conditions are consistently grouped together by the algorithm.

In conclusion, reducing the dimensions of the dataset helped in improving computational efficiency. While clustering showed some alignment with pleasant weather labels, there may still be room to fine-tune the clustering thresholds or explore additional clustering methods for more conclusive results.