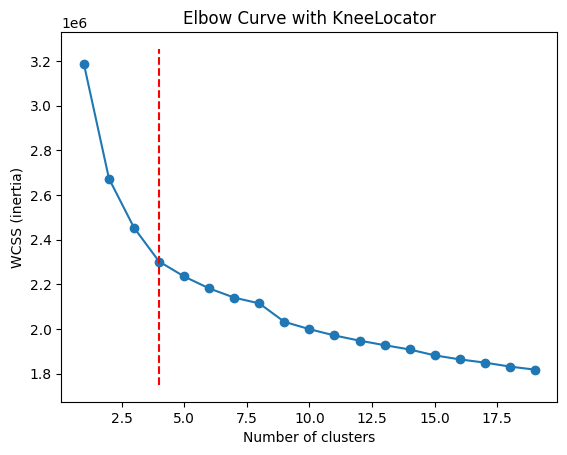
**Pump Equipment Sensor Data Anomaly Detection**

This project simulates sensor readings for various types of pump equipment, introduces anomalies and operator logs, and applies unsupervised machine learning techniques to detect anomalies.

1. **Data Creation:**

* Sensor Data Generation
  1. Equipment Types:
     1. Seawater Lift Pump
     2. Centrifugal Pump
     3. Mud Pump
  2. Sensor Design:
     1. Identified essential sensors (Temperature, Pressure, Vibration, etc.) for each equipment. Defined operational ranges based on domain knowledge.
  3. Granularity:
     1. Synthetic data generated per minute.
  4. Outlier Injection:
     1. High/Low Outliers added by configuring low and high outlier means to mimic real-world drift such as:
     2. Overheating due to auto-temp control failure
     3. Weather-driven fluctuations
     4. Injected in ~0.3% of the records.
  5. Missing Data:
     1. Introduced in ~0.5% of the values to simulate real-world data loss.
* Operator Log Generation
  1. Log Templates:
     1. Handwritten templates with varying structure and language were used to simulate human-written logs.
  2. Time Frequency:
     1. Logs generated every 30 minutes.
  3. Log Coverage:
     1. Each log tied to a specific Equipment ID and Sensor Type
     2. Special logs were inserted to correlate with injected anomalies

1. **Anomaly Detection**
   1. Strategy
      1. Anomalies were detected at equipment type level to reflect varying normal ranges.
      2. Timestamps were feature-engineered into:
      * Hour
      * Time in Minutes
      * Day of Week
      * Month
      * Year
   2. Algorithms Explored
      1. K-Means Clustering
         1. Clusters form around normal operational behavior; outliers fall outside the dense cluster zones.
         2. Chosen due to:
         * Well-separated cluster formation
         * Effective detection of subtle abnormal patterns
   3. KneeLocator used with Elbow Method to determine optimal number of clusters.



b. Isolation ForestUnsupervised technique that learns normal behavior and flags deviations.

* Helpful for sparse anomaly detection.

1. Final Model Selection

* K-Means was selected:
  + Showed clearer separation of anomalies from normal points
  + Produced anomaly groups with significantly higher/lower sensor values
  + Better interpretability for visual inspection
    1. Isolation Forest:
       1. This algorithm learns what “normal” looks like and flags rare patterns.
  1. K-Means cluster was chosen as the final model because the it highlighted the outliers and the anomalies in the data better than the Isolation forest. The mean of the anomalies highlighted in the K-means where much low and high as compared to the normal values.

