Let’s analyze **why the last red point in early 2024** is flagged as an **outlier**, even though it **doesn’t appear very extreme** visually compared to the surrounding values.

**🧠 Key Insight:**

The rolling Z-score **doesn't only depend on how high or low a point is globally**, but rather **how much it deviates from the *local mean and standard deviation*** at that point in time.

**🔎 Possible Explanation for the Outlier:**

Let’s break it down:

**✅ 1. Rolling Window Context Is Important**

Suppose your rolling window is, for example, the past **30 days**.

* If all the values in the recent 30-day window before that point are tightly grouped (e.g., between 60 and 65),
* And then suddenly, this last value drops to **40**, that **relative deviation** can trigger a Z-score below -3.

**✅ 2. Small Local Standard Deviation (σ)**

If the values in the rolling window before that point are very stable (low variance), then even a **small deviation** from the mean can result in a **high Z-score** in absolute value.

**✅ 3. Normalization Is Local, Not Global**

Even if the global dataset has wider swings (e.g., between 0 and 120), the rolling Z-score **only cares about the local window**.  
So what looks “normal” globally may still be a local outlier.

**🧪 Example (Hypothetical):**

Let’s say, in January 2024:

* Rolling mean (μ) = 65
* Rolling std deviation (σ) = 3
* Current value (xₜ) = 55

Then:

Z=55−653=−3.33Z = \frac{55 - 65}{3} = -3.33Z=355−65​=−3.33

Even though **55 is not very low globally**, the **local context says it's abnormal** → → **Outlier!**

**✅ Summary:**

The last red point is flagged because it deviates significantly from the recent local trend — not because it's globally extreme, but because it's **unexpected relative to the recent 30-day window (or whatever window size you set)**.

If you want to make this method less sensitive, you can:

* Increase the **rolling window size** (e.g., from 30 to 60 or 90 days),
* Or increase the **Z-score threshold** (e.g., from 3 to 4).

### So yes — those red dots are based on **the local behavior of the previous N points**, not global distribution. That's why even "normal-looking" points can become outliers if they break from a recent pattern

### نکته مهم:

الگوریتم **Z-Score متحرک (Rolling Z-Score)** فقط به بالا یا پایین بودن مقدار به‌صورت کلی نگاه نمی‌کند؛ بلکه **بررسی می‌کند آیا مقدار فعلی نسبت به مقادیر اخیر غیرعادی است یا نه.**

## خلاصه:

نقطه آخر قرمز در سال 2024 به این دلیل پرت (outlier) شناخته شده که **نسبت به داده‌های اخیر (نه کل داده‌ها)** تفاوت زیادی داره.  
حتی اگر در نگاه کلی طبیعی به‌نظر برسه، الگوریتم اون رو **در زمینه محلی خودش** غیرعادی تشخیص داده.

//////////////////////////////////////////////////////////////////////////////

for the series data like energy consumption which outlier can be used?z-score or IQR

Great question! When dealing with **time series data like energy consumption**, both **Z-Score** and **IQR** can be used for outlier detection — but they behave **very differently** depending on the data's nature.

Let’s compare the two and help you choose the **best method** 👇

## ✅ 1. ****Z-Score (Standard Score Method)****

### 📌 Best for:

* **Stationary** time series (where mean and std don't shift too much),
* Real-time detection using **rolling window** (rolling mean & std),
* Detecting **sudden spikes or drops**.

### 🧠 How it works:

* Measures how far a value is from the **local or global mean** (in terms of standard deviations).
* Can be made dynamic using a **rolling window**: detects deviations based on recent trend.

### ⚠️ Caution:

* Sensitive to outliers in small windows.
* Assumes **normal distribution**, which time series may not always follow.

### ✅ Good choice for:

✅ Energy usage patterns where values change gradually but **occasional spikes** or **dips** are meaningful (e.g., holiday peaks, sensor failures).

## ✅ 2. ****IQR (Interquartile Range Method)****

### 📌 Best for:

* **Non-time series** or **static datasets**,
* Datasets with **non-normal** distribution or **heavy tails**,
* Detecting outliers based on **overall distribution** rather than local context.

### 🧠 How it works:

* Calculates:

IQR=Q3−Q1\text{IQR} = Q3 - Q1IQR=Q3−Q1

* Outlier if:

x<Q1−1.5×IQRorx>Q3+1.5×IQRx < Q1 - 1.5 \times IQR \quad \text{or} \quad x > Q3 + 1.5 \times IQRx<Q1−1.5×IQRorx>Q3+1.5×IQR

### ⚠️ Caution:

* Not aware of **temporal structure** → doesn’t consider **recent trends**.
* Can miss **local anomalies** in time series.

### ✅ Good for:

✅ One-time **batch analysis** of entire consumption dataset to find **global extremes** (e.g., extremely low or high usage regardless of time).

## 🆚 Summary: Which One for Energy Consumption?

| **Method** | **Best For** | **Considers Time?** | **Real-Time Use** | **Detects Local Change?** |
| --- | --- | --- | --- | --- |
| **Z-Score (Rolling)** | Sudden spikes/drops | ✅ Yes | ✅ Yes | ✅ Yes |
| **IQR** | Global anomalies | ❌ No | ❌ No | ❌ No |

## جایگاه این روش در دسته‌بندی کلی روش‌های تشخیص Outlier:

طبق مقاله، روش‌های تشخیص ناهنجاری به سه دسته تقسیم می‌شوند:

| **نوع** | **مثال** | **توضیح** |
| --- | --- | --- |
| 1. روش‌های آماری (Statistical) | Z-Score, IQR, REHWS | فرضیه‌هایی روی توزیع داده دارند |
| 2. یادگیری ماشین سنتی (ML) | Isolation Forest, LOF | بدون فرض توزیع، یادگیری از الگو |
| 3. یادگیری عمیق (DL) | LSTM, AutoEncoder | یادگیری دنباله‌ای پیچیده |