Statistical Process Control

MSc: Statistics and Actuarial-Financial Mathematics

Konstantinos Bourazas
kbourazas@aegean.gr

Department of Statistics and Actuarial-Financial Mathematics

Course 1

February 19, 2026

Why statistical process control matters

- In many real systems, observations arrive **sequentially** rather than all at once
- We need to decide **as data come in** whether the underlying process remains stable or has **changed**
- Not every fluctuation is a problem
  - Some variation is expected even when the process is operating normally
  - Other variation reflects a real change that may require investigation or action
- The central task is to distinguish **common cause variation** from **special causes** early and reliably

Common cause vs special cause variation (I)

Definition

Common cause variation is the natural, routine variability that is built into the process when it operates normally.

- It comes from many small sources acting together, such as minor differences in raw materials, ambient conditions, or measurement noise
- Even with the same settings and the same operator, the output will not be identical from one time point to the next
- In Statistical Process Control we treat this variation as the **baseline** and we use it to build a statistical model for the process and set control limits

Common cause vs special cause variation (II)

Definition

Special cause variation is additional variability caused by a specific change or event that is not part of normal operation.

- It is linked to an identifiable reason, for example tool wear, calibration problems, a new supplier, a change in procedure, or a sensor fault
- It may affect the process level (mean), the variability (variance), or the overall shape of the distribution
- The goal of Statistical Process Control is to detect these departures from the In Control baseline quickly, while keeping false alarms rare

What is Statistical Process Control (SPC)?

Definition

Statistical process control is an effective area of Statistics that includes all methods that deal with the **quick and valid detection of any disorder** in an ongoing process.

- Its main aim is to detect when a process moves from an **In Control (IC)** state to an **Out of Control (OOC)** state
- SPC is implemented primarily via **control charts**, originally developed in industry (Shewhart, 1926) and now widely used across many domains
- There are charts for different data types such as **continuous measurements**, **counts**, **binary outcomes**, **correlated time series**, and **multivariate streams**

Reference Shewhart, W. A. (1926). *Quality control charts*. The Bell System Technical Journal, 5(4), 593–603.

Examples across domains

Typical domains and monitoring goals

- **Industrial processes**
  - Measurements such as product dimensions or weight, temperatures, machine outputs
  - **Goal:** detect tool wear, calibration drift, or sustained shifts in quality early enough to intervene
- **Markets and finance**
  - Time series such as asset returns, credit spreads, exchange rates, liquidity or risk indicators
  - **Goal:** detect regime changes, abnormal volatility, or structural shifts that affect risk and decisions
- **Infrastructure and sensor networks**
  - Streams such as pressure and flow in water networks, sensor signals in smart city systems
  - **Goal:** detect leaks, sensor malfunction, gradual drift, or sudden disruptions before they escalate
- **Health**
  - Sequential measurements such as lab values and patient monitoring in intensive care units
  - **Goal:** detect clinical deterioration or changes in patient state early enough to act

Types of shifts we aim to detect

The **Out of Control** states most often considered in practice can be grouped into two broad types:

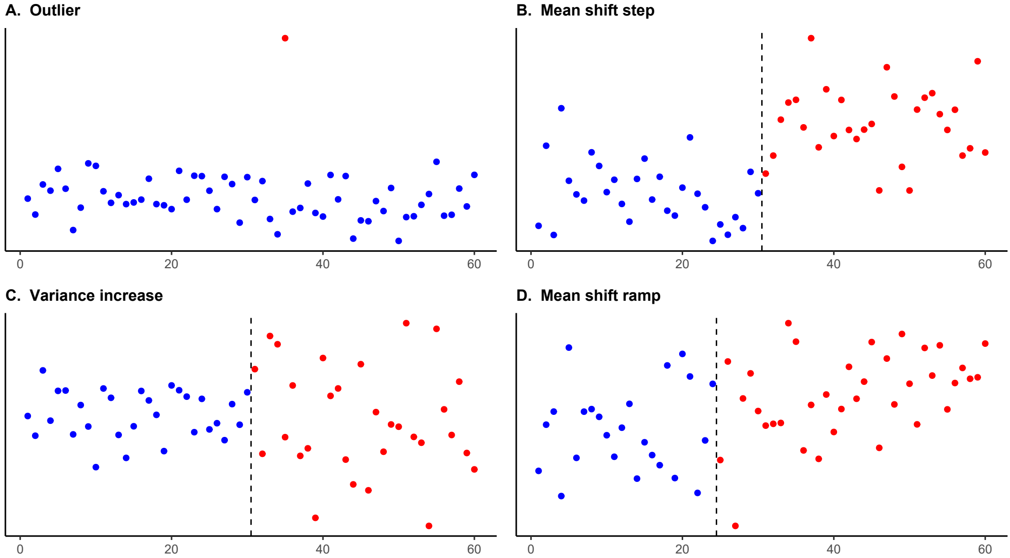
Transient shifts

- An isolated unusual value, that is an **outlier**
- Typically large in magnitude and short lived
- Often caused by a one off event such as a measurement glitch or a brief external disturbance

Persistent shifts

- Systematic changes that affect one or more aspects of the process, often in the **mean** and/or the **variance**
- Typically small to medium in magnitude but **sustained** over time
- Common forms include
  - **Step change** in the mean level
  - **Ramp change** or linear trend in the mean
  - **Scale shift** as an increase or decrease in variability

Four typical change patterns



Phase I and Phase II in SPC

- Most **SPC** methods for detecting **OOC** behavior are naturally organized into two phases
- The reason is simple
  - We first need a reliable description of the process when it is **IC**
  - Then we can compare new observations against the **IC** baseline and decide whether the process has moved to an **OOC** state

Phase I

Phase I is the training and typically offline phase, where independent **IC** data are gathered and the goal is to calibrate the monitoring scheme

Phase II

Phase II follows and it is the testing and typically online phase, where new observations are collected and compared against the **IC** standards established in Phase I

Phase I vs Phase II in practice

Phase I

- Select a time period where the process is believed to be stable
- Estimate the key **IC** characteristics such as the mean and variance
- Set control limits or other decision thresholds that define typical **IC** variation
- Review the historical data for obvious special causes and refine the baseline if needed
- Output is a set of calibrated **IC** standards ready for monitoring

Phase II

- Monitor new observations sequentially and compare them to the Phase I **IC** standards
- When the chart signals, treat it as evidence of **OOC** behavior and investigate the root cause
- The monitoring design balances quick detection of real changes with keeping false alarms rare

Control limits and alarms

- In **SPC**, we monitor a process by plotting a statistic over time
  - The statistic can be a measurement, a subgroup summary, a count, or a smoothed score
  - It is designed to be stable when the process is **IC** and to react when the process becomes **OOC**
- We set **control limits**: an upper and a lower boundary around the expected IC behaviour
  - If the plotted statistic crosses a control limit, we **raise an alarm** (evidence of **OOC**)
  - If it stays within the limits, we treat the data as consistent with **IC**
- Different control charts correspond to different choices of the plotted statistic

False alarms and calibration of control limits

- A **false alarm** occurs when the chart signals **OOC** even though the process is actually **IC**
  - It triggers unnecessary investigation and wastes resources
  - Too many false alarms reduce trust in the scheme and may lead to real alarms being ignored
- Control limits must be calibrated so that false alarms are **rare under IC**
  - This calibration is done in **Phase I**, using data that represent IC operation
  - The limits are placed so that typical IC variation stays inside them most of the time
- There is an unavoidable trade-off, directly linked to hypothesis testing:
  - **Tighter limits** increase sensitivity but also the **false alarm rate**, which is essentially a Type I error since we conclude OOC when the process is actually IC
  - **Wider limits** reduce false alarms but increase the **missed detection rate**, which is essentially a Type II error since we fail to detect a real shift
  - Good chart design balances these two errors for the application at hand

Control limits vs specification limits

Control limits

Control limits are statistical decision boundaries used in **SPC**

- They are estimated in **Phase I** using **IC** data
- They define the range where the charted statistic is expected to fall when the process is **IC**
- Crossing a control limit is evidence for **OOC** and triggers an investigation

Specification limits

Specification limits are requirements for the product or service

- They are set by engineering design, contracts, regulations, or customer expectations
- They define what values are acceptable, regardless of whether the process is **IC** or **OOC**

- A process can be **IC** but still produce values outside specifications
- A process can also be **OOC** even if most outputs remain within specifications

Rational subgrouping

- Many control charts are built on **subgroups** of observations collected close in time
- A **rational subgroup** is formed so that variation **within** the subgroup reflects common causes only
  - Observations inside the same subgroup should be as similar as possible
  - Differences **between** subgroups should capture changes in the process
- This is a design choice that affects what the chart can detect
  - Poor subgrouping can hide real shifts or create unnecessary alarms
  - Good subgrouping increases sensitivity to meaningful changes
- Typical examples include consecutive items from the same machine setting, or sampling at fixed times within each shift

How often should we sample

- There is no universal rule for the sampling interval
- The sampling plan is a design choice based on practical trade offs
  - How quickly the process can drift or fail
  - The cost of delayed detection versus the cost of sampling and inspection
  - The production rate and the amount of output produced between samples
- A useful way to think about it
  - Subgroups should be collected so that within subgroup variation reflects common causes
  - The time between subgroups should be short enough to detect important shifts before too much output is affected
- In practice, initial sampling plans are often refined after learning the process and reviewing performance