Exploratory Data Analysis (EDA)

This document captures what I have learned about Exploratory Data Analysis (EDA) through hands-on work, watching videos, using AI, etc. The emphasis is on thinking, judgment, and decision-making, not just plots.

Exploratory Data Analysis (EDA) is frequently misunderstood as a mere preliminary chore, a quick cycle of generating "pretty plots" to satisfy a project requirement. Instead, EDA is a rigorous process of investigative journalism applied to a dataset. Its primary mission is the systematic reduction of uncertainty, serving as the compass that answers the most critical question in any technical project: “What should I do next?”

At the onset of any data project, the researcher exists in a state of profound ignorance. We do not yet know if the data is a faithful representation of reality or a collection of artifacts born from broken sensors and logging errors. We cannot assume that the relationships between variables are linear or even meaningful; often, the most prominent patterns are spurious, leading toward conclusions that dissolve under scrutiny. EDA exists to dismantle this uncertainty. It forces the analyst to confront the data's integrity, identifying where "zero" values are placeholders for missing information and where extreme outliers represent either a breakthrough insight or a fundamental failure in data collection.

To perform EDA effectively, one must move beyond the "statistical dump"—the habit of running every T-test or correlation matrix available without a guiding hypothesis. A correlation coefficient of zero, for instance, does not necessarily imply the absence of a relationship; it may simply hide a complex, non-linear trend that a standard test is blind to. By using visualization as a diagnostic tool rather than an aesthetic one, the analyst can see the "geometry" of the data. This allows for a pivot from blind modelling to informed decision-making. Whether the next step is to clean the data, transform a variable, or pivot to a different modelling architecture entirely, that choice is rooted in the empirical evidence uncovered during exploration.

Ultimately, EDA is a discipline of scepticism. It is the phase where you stop looking at what the data *should* say and start listening to what it is actually saying. By prioritizing the reduction of uncertainty over the generation of visual artifacts, the analyst ensures that the foundation of their project is secure. Everything in the EDA toolkit, i.e. grouping, slicing, and visualizing, is in service of clarity. It transforms a data scientist from a passive observer into a strategic architect, ensuring that every subsequent step is not a shot in the dark, but a calculated move toward a reliable truth.

The most dangerous trap a data scientist can fall into is the belief that Exploratory Data Analysis is a checklist of code blocks. In this "command-driven" mindset, the analyst is merely a technician performing a ritual, hoping that if they run enough scripts, an insight might accidentally fall out of the screen. But true EDA is not a coding exercise; it is a high-stakes **iterative reasoning loop**.

When you move from a coding mindset to a thinking mindset, your primary tool is no longer the library you import, but the curiosity you bring to the keyboard. This shift requires moving from "What can I plot?" to "What do I need to understand?" In a professional environment, good EDA often looks deceptively slow. A peer reviewing your work might see you spend three hours on a single subset of data, pivoting and filtering it a dozen different ways. To the untrained eye, this looks like spinning wheels. In reality, it is the most productive work you can do. By ruthlessly interrogating the data early on, you prevent the catastrophic time-sink of building an elaborate model on a foundation of misunderstood variables.

The "Thinking" mindset follows a specific, human rhythm. It begins with an **observation**: perhaps a strange spike in a time-series or an unexpected cluster in a scatter plot. A technician notes the spike and moves to the next chart; a thinker asks, **"Why is that there?"** This leads to the most critical stage of the loop: **checking alternative explanations**. If sales spiked in July, was it a successful marketing campaign, or did the data engineering team accidentally double-count transactions during a server migration?

By refining the question through this process of elimination, you systematically reduce the "search space" of uncertainty. **This loop-Observe, Ask, Check, Refine**, is what separates a meaningful discovery from a superficial observation. It transforms EDA from a static report into a dynamic, living conversation with the data. When you stop treating your scripts as a series of instructions and start treating them as a telescope, you begin to see the hidden structures that actually define the problem. Ultimately, the goal is to ensure that you aren’t just guessing, you’re executing a strategy built on hard-won clarity.

Before diving into analysis, you must treat the dataset like a crime scene that needs to be surveyed. This "First Contact" requires answering four fundamental questions that define the boundaries of your project:

1. What does one row represent?

This is the "grain" of your data. Is a single row one customer, one transaction, or one second of sensor data? If you mistake a dataset of *daily averages* for a dataset of *individual events*, your entire understanding of variance and distribution will be fundamentally flawed. Defining the row is the only way to ensure you are counting the right things.

2. What is the unit of each variable?

Ambiguity is the enemy of insight. A column labelled "Temperature" is useless unless you know if it is Celsius, Fahrenheit, or Kelvin. A "Price" column could be in USD, cents, or a currency that no longer exists. Analysing data without confirmed units isn't just a mistake—it’s a liability.

3. What are the implicit assumptions?

Data is rarely "raw." It has usually been processed, aggregated, or filtered before it reaches your desk. You must hunt for the invisible logic: Is this station-level aggregation? Are these figures adjusted for inflation? Are "null" values actually zeros? Identifying these assumptions prevents you from "discovering" a trend that was actually hard-coded into the data by the person who exported it.

4. What is *not* measured?

The most important variable in your project might be the one that is missing. If you are analysing retail sales but don't have a variable for "Out of Stock" events, you might mistake a supply chain failure for a drop in consumer demand. Acknowledging the "known unknowns" keeps your analysis grounded in reality rather than speculation.

Importance of Cleaning

There is a persistent myth in data science that "cleaning" and "exploring" are two separate phases, separated by a distinct boundary. The standard workflow is often described as a linear assembly line: first, you scrub the data of errors, and only then do you begin the "real" work of

Exploratory Data Analysis (EDA). This view is not just wrong; it is dangerous. In reality, data cleaning is not a preliminary chore, it is, in itself, a form of analysis.

The problem with treating cleaning as a mindless pre-step is that you cannot effectively clean what you do not understand. How can you decide if a data point is an error or a critical anomaly without first plotting its distribution? If you blindly apply a rule, such as deleting all rows with missing values or clipping data beyond three standard deviations you are not cleaning the data; you are censoring it. You are making decisions about the reality of your dataset before you have even looked at it.

Effective cleaning is deeply intertwined with EDA. It is a cycle, not a sequence. For example, consider a dataset from physical sensors. You might find a variable that spikes wildly for three seconds. Is this a system failure or a genuine event? You cannot know until you visualize the time-series context (EDA). If you see the spike correlates with a known system reboot, you can confidently clean it. Without that visual context, deleting it is just a guess.

Similarly, consider missing data. A novice's approach might simply drop all rows with null values to get the code to run. But EDA requires you to visualize *where* those values are missing. Visualizing these missing values can add a lot of insight to your analysis. Did people deny to give a survey? Who were those people, what was their incentive to deny the survey, etc. Analysing this would expose the hidden bias of your dataset, which would otherwise have been ignored by a novice.

Univariate Analysis

When you first sit down with a new dataset, the natural impulse is to jump straight into summary statistics:- mean, median, standard deviation, maybe the min and max.

Univariate analysis is really about getting to know one variable at a time with genuine curiosity. You're asking three simple but powerful questions: What values are typical? What values are rare? And is the distribution symmetric or skewed? Answering these properly almost always requires looking at the data visually, not just crunching it into a handful of scalars.

Start with the basics. A histogram shows you the actual shape of the distribution. You can see right away if the data clusters around a central value (symmetric, like a nice bell curve), stretches out to one side (skewed right or left), or has multiple peaks (multimodal, hinting that there might be distinct subgroups hiding in what looks like one variable). A single mean or median tells you almost nothing about those patterns. The classic example is Anscombe's quartet: four completely different datasets with identical means, variances, and correlation, until you plot them and realize one is a straight line, another is mostly linear with one wild outlier, and so on. The numbers don't warn you; the picture does.

Boxplots are another important . They give you a quick snapshot of spread and symmetry through the median, interquartile range (IQR), and whiskers, while flagging potential outliers. They're especially useful when you have dozens of variables and want to compare them side-by-side without drowning in histograms. But even the boxplot has a limit. It summarizes without showing the full density. That's why I almost always pair it with a histogram or a kernel density plot. The box tells you where the middle 50% lives and how far the tails reach; the histogram reveals whether that middle is a gentle mound or a sharp spike.

Relying solely on summary statistics is dangerous precisely because they hide so much. The mean gets dragged around by extreme values (think billionaire incomes skewing average wealth in a neighborhood). The median can stay calm in the middle while the data actually has two clear clusters on either side—multimodality that completely changes how you interpret the variable. Variance or standard deviation tells you about spread but nothing about shape. Without visuals, you're basically navigating with your eyes closed.

In practice, I always follow the same flow for univariate work: compute the five-number summary (min, Q1, median, Q3, max) plus mean and maybe mode, then immediately plot a histogram and boxplot. Only after seeing both do I trust what the numbers are telling me. Sometimes the plot reveals a need to transform the variable (log for right-skewed positive data), bin categories differently, or even question whether the data was recorded correctly.

Respecting each variable means refusing to reduce it to a few tidy stats before you've really looked at it. The plots aren't optional extras—they're the primary way you discover what the data is actually saying. Skip them, and you risk building everything that follows on a foundation of convenient fictions. In data analysis, seeing really is believing.

Bivariate Analysis

Bivariate exploratory data analysis (EDA) is often treated as a mechanical step: take every variable, pair it with every other variable, and compute correlations or scatter plots. While this approach is easy to automate, it rarely leads to understanding. Meaningful bivariate analysis begins not with plots, but with questions.

The first question to ask is whether a relationship is physically or logically plausible. Data does not exist in isolation, it represents real processes. For example, the relationship between AQI and PM2.5 is worth examining because AQI is partly constructed from pollutant concentrations, and PM2.5 is a known driver of poor air quality. Similarly, AQI versus wind speed makes sense because wind affects dispersion of pollutants. A negative relationship here is not surprising and is grounded in atmospheric behaviour.

In contrast, pairing AQI with station ID does not convey meaningful information on its own. Station ID is an identifier, not a physical quantity. Any apparent relationship would be an artifact unless spatial effects, clustering, or regional fixed effects are explicitly modelled. Plotting such pairs without context wastes effort and risks misleading interpretation.

The second key question is whether two variables are proxies for the same underlying phenomenon. Many environmental variables are strongly correlated because they co-occur or are generated by related processes. For instance, PM2.5 and PM10 often rise together during pollution episodes. Examining both against AQI is useful, but interpreting them as independent explanatory signals can be misleading. In such cases, bivariate plots should be used to understand redundancy, not causation.

This perspective changes the goal of bivariate EDA. The aim is not to find “strong correlations,” but to test expectations derived from domain knowledge. When a plausible relationship behaves unexpectedly, that result is informative. It may indicate data quality issues, missing variables, seasonal effects, or non-linear dynamics that deserve deeper analysis.

Ultimately, good bivariate analysis is selective. Domain knowledge acts as a filter, narrowing attention to relationships that can be meaningfully interpreted. This discipline saves time, reduces noise, and leads to insights that are grounded in reality rather than statistical coincidence.

Outliers

Outliers often trigger a reflexive response: remove them. This instinct comes from a desire to clean data quickly and make models behave better. However, in exploratory data analysis (EDA), removing outliers without investigation risks discarding the most informative parts of the dataset.

The correct first step is not removal, but inquiry. An outlier is a question, not a problem. The task of EDA is to determine what kind of question it is asking.

The first check is physical plausibility. Does the value violate known limits or laws? For example, a negative pollutant concentration or an impossible temperature may point to a sensor or logging error. Such cases can often be justified for removal once confirmed.

If the value is physically possible, the next question is context. Does the outlier occur during specific conditions such as certain seasons, weather patterns, or times of day? In air quality data, extreme AQI values may coincide with stagnant wind conditions, temperature inversions, or festival-related emissions. These points are not noise; they are expressions of the system under stress.

Another important question is alignment with known events. Sudden spikes may correspond to fires, dust storms, industrial incidents, or policy changes. Without cross-checking against timelines or external knowledge, these signals can be misclassified as anomalies and removed incorrectly.

Outliers generally fall into three categories: sensor failures, rare but real phenomena, and high leverage observations. Sensor failures should be handled or excluded. Rare events should be preserved and understood. High-leverage points, even if few in number, often dominate outcomes and explain why averages fail to capture reality.

EDA is the stage where these distinctions are made. The goal is not to force the data into neat distributions, but to learn how the system behaves at its extremes. In many real-world datasets, the outliers are not mistakes, they are the reason the data is worth studying.

Asking Better Questions Through EDA

Exploratory data analysis is often described as a way to “find answers” in data. In practice, it is more important role is to improve the quality of questions being asked. Good EDA rarely produces final conclusions, instead, it refines vague curiosity into precise, testable hypotheses.

Early questions are usually broad and underspecified. For example, asking “Which pollutant is highest?” treats pollution as a static quantity and ignores context. Through EDA, this question evolves into something more meaningful: “Which pollutant dominates AQI during extreme days versus normal days?” This shift introduces conditions, comparisons, and relevance. It recognizes that averages hide behavior at the tails, where impact is often greatest.

Similarly, the question “Does weather affect AQI?” is too coarse to guide analysis. Weather is not a single factor, and its effects are rarely uniform. EDA encourages a more structured question: “Which weather variables matter, and how does their influence change with season?” This framing acknowledges interaction effects and avoids assuming a single global relationship.

This evolution happens because EDA exposes patterns, inconsistencies, and structure in the data. Seasonal clustering, regime changes, and non-linear relationships become visible before any formal modelling begins. Each plot or summary either confirms an expectation or challenges it, forcing the analyst to be more specific.

The value of EDA lies in this sharpening process. It narrows the problem space, reduces ambiguity, and prevents premature modelling based on poorly formed questions. By the time formal analysis begins, the questions are grounded in observed behaviour rather than intuition alone.

In this sense, EDA is not a preliminary step to be rushed through. It is the stage where thinking improves. Better questions lead to better models, and EDA is the tool that makes that progression possible.

Common Mistakes I Learned to Avoid

The most common EDA mistakes come from treating it as a mechanical task rather than a thinking process. Plotting without a clear question produces visuals that look informative but explain nothing. Blindly trusting correlation ignores causality, confounding, and shared drivers. Ignoring seasonality collapses distinct regimes into misleading averages.

Over-cleaning removes rare but meaningful behaviour in the name of neatness. Treating EDA as a checklist encourages completion over understanding. Each of these mistakes creates confidence without insight, leading to conclusions that feel rigorous but are fundamentally wrong.