Is Your Data AI-Ready?

On Domo's AI Labs team, our data scientists excel in weaving AI strategies into business operations. A significant part of their day? Ensuring the data feeding into algorithms is pristine. Because, at the end of the day, the strength of your AI depends on the quality of your data.

No algorithm can compensate for poor data quality. Instead of investing in expensive fixes later, why not get it right from the start? Our team has encountered numerous data challenges that we've overcome, and now, we're sharing a checklist to help you sidestep common pitfalls.

Below, you'll find essential tips for cleaning your data—making sure it's polished and primed for any AI application you envision.

LABELING

Check	Goal	Example
☐	Data in columns must match column labels.	Column with revenue data should include "revenue" in the column name.
☐	Data column labels are unique (do not match any other column names) and concise/short, yet sufficiently intuitive for understanding what the data in the column represents.	*Emails_Opened_Last_Year* is a concise label for a column containing data on emails opened in the last year.
☐	Data column descriptions are included when necessary or helpful, particularly if intending to use generative AI to explore the data set.	If a column is labeled *Revenue*, but only includes the revenue from certain product lines, this should be noted in the description (or better yet, indicated in the column label).
☐	Columns are appropriately described in the data set when values in a column are not intuitive.	"K-product sales" is an example of a nonintuitive data value (what is a K-product?).

ROW CONSIDERATIONS

Check	Goal	Example
☐	There is a unique identifier/primary key for each row, meaning a column or columns that contain values that uniquely identify each row in a data set. The unique identifier column(s) should be the first column(s) in the data set.	*EMPLID*, a column containing the unique ID number for each employee at a company, is usually the unique identifier/primary key in a data set containing employee data.
☐	Note in the data set description about what the data granularity is (what does a row represent?) and which columns in the data set are unique identifier(s).	Example data set description: *Each row corresponds to an employee in the company. EMPLID is the unique identifier.*
☐	Rows/observations are reasonable to include in data set for analysis and others are removed.	Remove rows for "admin", "test", "internal", etc.
☐	Duplicate rows are identified, flagged, and managed programmatically.	Rows that have the same value across all columns.
☐	Rows with duplicate identifiers are identified, flagged, and managed programmatically.	Rows that may not have the same value across all columns but that have the same unique identifier.

WITHIN-COLUMN ACCURACY & CONSISTENCY

Check	Goal	Example
☐	Data type is appropriate given the values in a column.	Values in a column that are numeric are stored as a fixed decimal, floating decimal, or integer data type; text values are stored as a string data type; dates and times are stored as date or time-stamp data type.
☐	Data schema is consistent.	Phone numbers are 10 digits with no special characters or spaces; states are represented with the correct two-letter abbreviations.
☐	Range of data values in each column is appropriate and accurate.	All data values for a given column fall within an expected and specified range; e.g., a column with data on people's ages should generally not include any negative values or values greater than 110.
☐	Outliers are identified, flagged, and managed programmatically for each column.	Unusually high and low data points in each column that look suspicious must be investigated and appropriately managed.
☐	Categorical variables are clean and categories are grouped following conceptual/business/statistical logic.	Buckets are combined based on logic.
☐	Number of buckets for categorical variables are reasonable for analysis.	Are there too many buckets considering the amount of data? Can smaller buckets be combined?
☐	Bucket size for categorical variables is reasonable for analysis.	Buckets that describe small percentages of the rows in the data set should be managed.
☐	Distribution of data in each column is understood, and necessary data transformations should be completed.	Normal, skewed, other, etc.
☐	Missing data is coded consistently within each column.	Placeholder values for missing data, "N/A", empty cells, or other methods for identifying missing data should be applied consistently.
☐	Missing data is identified, flagged, and managed programmatically (many data science algorithms automatically exclude any row with missing data in at least one cell; missing data generally needs to be managed to avoid this).	Although we may have certain features, how well are these features defined? Is the majority of data defined or missing? Can we compute missing data?
☐	Provide formula used for calculation of the metric, and verify that the implementation matches the calculation.	If profit is revenue minus COGS, verify that it is implemented in a Beast Mode.

BETWEEN-COLUMN ACCURACY & CONSISTENCY

Check	Goal	Example
☐	Appropriate granularity is applied or labeled, such that the same unit of analysis is expressed in each column of the data set.	Hourly, daily, weekly, monthly, etc.
☐	Appropriate aggregations are applied or labeled, such that the same unit of analysis is expressed in each column of the data set.	Uniques, counts, averages, etc.
☐	Duplicate columns are identified, flagged, and managed programmatically.	Are they truly duplicate columns?
☐	Constants are identified, flagged, and managed programmatically.	Columns with only one value are generally not useful in analyses.
☐	Other data and calculation errors are identified, flagged, and managed programmatically.	Beast Modes, ETLs, etc.
☐	Missing data has been explored and patterns of missing data across columns is understood.	Do certain rows have missing data consistently across the same features?

VERIFICATION OF JOINS

Check	Goal	Example
☐	Row count of data sets must be validated both before and after joins.	If a table is left joined, it should only include up to the number of rows as in the left table.
☐	Join keys must be cardinal.	If you are joining a customer and orders data set, does each customer ID truly correspond to just one customer in the orders data set?