A **comprehensive chart** listing essential **functions and methods used in data analysis** with libraries like **NumPy**, **Pandas**, **Matplotlib**, and **Seaborn**. It includes categories for better clarity: **data inspection, cleaning, manipulation, aggregation, visualization**, etc.

**🔍 DATA ANALYSIS FUNCTION/METHODS CHART**

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | **Function/Method** | **Library** | **Purpose / Description** |
| **Data Loading** | read\_csv() | pandas | Load CSV files |
|  | read\_excel() | pandas | Load Excel files |
|  | read\_json() | pandas | Load JSON files |
|  | loadtxt(), genfromtxt() | NumPy | Load numerical data from text files |
|  | np.load() | NumPy | Load .npy binary files |
| **Basic Inspection** | .head(n) / .tail(n) | pandas | View top/bottom rows |
|  | .info() | pandas | Summary of DataFrame |
|  | .describe() | pandas | Summary statistics |
|  | .shape, .columns, .index | pandas | Dimensions and labels |
|  | type(), df.dtypes | built-in / pandas | Data types |
| **Missing Data** | .isnull(), .notnull() | pandas | Detect missing values |
|  | .dropna(), .fillna() | pandas | Drop or fill missing data |
| **Filtering & Selection** | df[col], df[['a','b']] | pandas | Select columns |
|  | df.loc[], df.iloc[] | pandas | Label-based / index-based selection |
|  | Boolean Indexing | pandas | Conditional filtering |
| **String Handling** | .str.contains(), .str.replace() | pandas | String manipulation |
|  | .str.lower(), .str.upper(), .str.strip() | pandas | Case and trim |
|  | .str.extract(), .str.split() | pandas | Regex or split strings |
| **Datetime Handling** | pd.to\_datetime() | pandas | Convert to datetime |
|  | .dt.year, .dt.month, .dt.day | pandas | Extract date parts |
|  | .resample() | pandas | Time-series resampling |
| **Data Transformation** | .apply(), .map() | pandas | Row/element-wise operations |
|  | np.where(), np.select() | NumPy | Conditional transformation |
|  | .replace(), .astype() | pandas | Replace values / change type |
| **Math Operations** | np.mean(), np.std() | NumPy | Mean, std, etc. |
|  | .sum(), .mean(), .count() | pandas | Column-wise aggregation |
|  | .cumsum(), .cumprod() | pandas | Cumulative sums/products |
| **GroupBy & Aggregation** | .groupby() | pandas | Group by one or more columns |
|  | .agg() | pandas | Apply aggregation functions |
|  | .pivot\_table() | pandas | Create pivot tables |
| **Sorting** | .sort\_values() | pandas | Sort by column values |
|  | .sort\_index() | pandas | Sort by index |
| **Merging & Joining** | pd.merge() | pandas | SQL-like joins |
|  | pd.concat() | pandas | Concatenate DataFrames |
|  | .join() | pandas | Join columns using index/key |
| **Exploding** | .explode() | pandas | Expand list-like column to rows |
| **Duplicates** | .duplicated(), .drop\_duplicates() | pandas | Identify/remove duplicates |
| **Value Counts** | .value\_counts() | pandas | Count unique values |
|  | .nunique() | pandas | Count unique elements |
| **Indexing** | .set\_index(), .reset\_index() | pandas | Set/reset index |
| **Renaming** | .rename() | pandas | Rename columns/index |
| **Column Operations** | df['new'] = df['a'] + df['b'] | pandas | Create new columns |
|  | df.eval() | pandas | Evaluate string expressions |
| **Visualization - Line, Bar, Pie** | plt.plot(), plt.bar(), plt.pie() | matplotlib | Basic plots |
|  | sns.lineplot(), sns.barplot() | seaborn | Advanced plots with stats |
| **Visualization - Dist & Count** | sns.histplot(), sns.distplot() | seaborn | Distribution plots |
|  | sns.countplot() | seaborn | Count by categories |
| **Visualization - Box/Violin** | sns.boxplot(), sns.violinplot() | seaborn | Show distributions |
| **Visualization - Scatter/Pair** | plt.scatter(), sns.scatterplot() | matplotlib/seaborn | Scatter plots |
|  | sns.pairplot() | seaborn | Pairwise plots |
| **Heatmaps & Correlation** | df.corr() | pandas | Compute correlation matrix |
|  | sns.heatmap() | seaborn | Visualize correlation matrix |
| **Styling Tables** | .style.format(), .style.highlight\_null() | pandas | Style DataFrames |
| **Saving Data** | df.to\_csv(), df.to\_excel() | pandas | Save files |
|  | np.save(), np.savetxt() | NumPy | Save NumPy arrays |
| **Reshaping** | .melt() | pandas | Wide to long format |
|  | .pivot() | pandas | Long to wide |
|  | .stack(), .unstack() | pandas | Hierarchical reshape |

**✅ Example Handy Functions by Library**

**🔹 NumPy**

* np.array(), np.mean(), np.std()
* np.reshape(), np.arange()
* np.where(condition, a, b)
* np.isnan(), np.any(), np.all()

**🔹 Pandas**

* df.isnull().sum()
* df['col'].str.contains("abc")
* df.groupby('col')['val'].mean()
* df.pivot\_table(index='A', columns='B', values='C')

**🔹 Seaborn**

* sns.set\_style("whitegrid")
* sns.histplot(df['col'])
* sns.boxplot(x='category', y='value', data=df)
* sns.heatmap(df.corr(), annot=True)

**🔹 Matplotlib**

* plt.figure(figsize=(10,6))
* plt.title(), plt.xlabel(), plt.ylabel()
* plt.legend(), plt.grid(), plt.show()

**🧠 Tips**

|  |  |
| --- | --- |
| **Concept** | **Explanation** |
| lambda | Anonymous function, useful in .apply() |
| explode() | Converts list-like values into separate rows |
| str accessor | Applies string methods to Series |
| dt accessor | Applies datetime methods |
| map() vs apply() | map() is element-wise; apply() works on rows/cols |
| value\_counts(normalize=True) | Gives proportions instead of raw counts |