DataAnalyzer Technical Documentation

Table of Contents

1. Introduction

DataAnalyzer is a powerful tool designed for data scientists and analysts to efficiently process and visualize large data sets. This document provides comprehensive guidance on installing, configuring, and using DataAnalyzer.

2. System Requirements

Before installing DataAnalyzer, ensure your system meets the following requirements:

- **Operating System**: Windows 10 or later, macOS 10.15 or later, or a Linux distribution (Ubuntu 20.04 or later)
- **RAM**: Minimum 8 GB (16 GB recommended)
- **Disk Space**: At least 500 MB of free disk space
- **Python**: Version 3.8 or later

3. Installation

Windows

1. Download the installer from the DataAnalyzer Download Page.
2. Run the downloaded `.exe` file.
3. Follow the on-screen instructions to complete the installation.
4. Verify the installation by opening Command Prompt and typing:

```
dataanalyzer --version
```

macOS

1. Download the `.dmg` file from the DataAnalyzer Download Page.
2. Open the downloaded file and drag the DataAnalyzer app to the Applications folder.
3. Open Terminal and type:

```
dataanalyzer --version
```

Linux

1. Open a terminal window.

2. Use the following command to install via pip:

```
pip install dataanalyzer
```

3. Verify the installation:

```
dataanalyzer --version
```

4. Configuration

After installation, you may need to configure DataAnalyzer for optimal performance:

1. **Configuration File**: Locate the `config.yaml` file in your installation directory.
2. Modify settings such as:
   - Data file paths
   - Default visualization preferences
3. Save your changes and restart DataAnalyzer.

5. Usage

Basic Commands

To analyze a dataset, use the following command:

```
dataanalyzer analyze --file path/to/your/data.csv
```

Visualization

To generate a visualization:

```
dataanalyzer visualize --file path/to/your/data.csv --type bar
```

6. API Reference

Analyze Method

```
def analyze(file_path: str) -> dict:
    """
    Analyzes the data in the specified file.

    Parameters:
    file_path (str): Path to the data file.

    Returns:
    dict: Analysis results.
    """
```

Visualize Method

```
def visualize(file_path: str, chart_type: str) -> None:
    """
```

```
    Visualizes the data in the specified file.

    Parameters:
    file_path (str): Path to the data file.
    chart_type (str): Type of chart to generate (e.g., 'line', 'bar').
    """
```

7. Troubleshooting

- **Installation Issues**: Ensure you have the correct Python version installed.
- **File Not Found Errors**: Double-check the file path you provided.

8. Frequently Asked Questions (FAQs)

Q: Can I use DataAnalyzer on older versions of Windows?
A: DataAnalyzer requires Windows 10 or later.

Q: How can I update DataAnalyzer?
A: Use the following command:

```
pip install --upgrade dataanalyzer
```

9. Contact Information

For support or inquiries, please contact the DataAnalyzer support team at support@dataanalyzer.com.

```

This format maintains clarity and structure, making it easy for users to navigate the documentation! Let me know if you need further modifications or additional sections.
```