UV–Vis Spectroscopy Data Processing using Python

Overview

This repository provides a reproducible Python workflow for processing and analyzing UV–Vis spectroscopy data. It includes baseline correction, signal smoothing, and automated peak detection, making it applicable to a variety of chemical spectroscopic datasets. The project also showcases my Python programming skills applied to UV–Vis data, emphasizing learning data processing, visualization, and peak detection rather than performing precise quantitative analysis.

Problem Statement

Raw UV–Vis spectra often suffer from:

- Baseline drift
- Instrumental noise
- Overlapping absorption features

These effects can obscure chemically meaningful information and reduce analytical reliability. This workflow addresses such challenges using **well-established numerical methods** commonly applied in spectroscopy and analytical data processing.

Features

- Automated baseline correction using **Asymmetric Least Squares (ALS)**
- Noise reduction via **Savitzky–Golay** smoothing
- Robust peak detection based on **signal prominence**
- Export of detected peak positions and intensities
- Clean, publication-style visualization

Methodology

1. Data Input

- CSV file containing:
    - Column 1: Wavelength (nm)
    - Column 2: Absorbance (a.u.)

2. Baseline Correction

- **Asymmetric Least Squares (ALS)** algorithm
- Effectively removes background drift without distorting peak shapes

3. Signal Smoothing

- **Savitzky–Golay filter** applied to baseline-corrected data
- Preserves spectral features while reducing noise

4. Peak Detection

- Peaks identified using **SciPy's find_peaks**
- Detection criteria include:
    - Relative prominence
    - Minimum peak separation

5. Output

- Text file containing detected peak wavelengths and absorbance values
- Processed UV–Vis spectrum with annotated peak positions

Applications

- UV–Vis spectroscopy data preprocessing
- This project is intended for **educational purposes only** and is **not suitable for precise scientific analysis.**

Requirements

- Python 3.x
- NumPy
- Pandas
- SciPy
- Matplotlib

Data Ethics & Usage

- This repository contains Python scripts for data analysis developed as part of my learning and research activities. For confidentiality and proprietary reasons, **original experimental data are not shared online**. All example data included here are **simulated, anonymized, or publicly shareable**.
- No unpublished, proprietary, or supervisor-owned experimental data are included.

Author

Ahmed Abdelrahim
M.Sc. Student in Analytical Chemistry
Health of Sciences Institute
Sivas Cumhuriyet University, Turkey

Disclaimer

This code is intended for academic and research purposes. Users should adjust processing parameters according to their specific experimental conditions.