A User-Friendly Jupyter Notebook for Simplified Differential Pulse Voltammetry Analysis of Dihydroxy Phenols

Aravindan Natarajan* and Preethi Sankaranarayanan

Email: anatarjank@gmail.com* and preethiusda@gmail.com

Abstract

Differential pulse voltammetry (DPV) is a well-established electrochemical technique widely employed for quantitatively determining diverse analytes, particularly within complex matrices. However, the analysis of DPV data, especially in scenarios involving the simultaneous detection of multiple species, can be a complex and time-intensive undertaking. This article introduces a Jupyter Notebook developed to facilitate the efficient and streamlined analysis of DPV data acquired from the simultaneous detection of dihydroxy phenols. The notebook offers a user-friendly interface encompassing data import, pre-processing, peak identification, quantification, and visualization functionalities. By automating several key analysis steps, this tool reduces manual effort while simultaneously enhancing the accuracy and reproducibility of the process. Consequently, it offers particular value to researchers engaged in environmental monitoring, food analysis, and related disciplines where the detection of dihydroxy phenols is of critical importance. The complete source code for this tool is freely available and accessible via the following repository: https://github.com/anatarajank/Electrochemical-Sensor-Data-Analyzer

1. Introduction

Dihydroxyphenols (DHPs) are a class of organic compounds frequently encountered in diverse environmental matrices and industrial effluents[1]. Their presence poses significant health risks owing to their inherent toxicity and potential carcinogenic properties[2]. Consequently, the accurate and sensitive detection and quantification of DHPs are of paramount importance for both environmental monitoring and rigorous quality control.

Electrochemical techniques, particularly differential pulse voltammetry (DPV), have emerged as powerful analytical tools for DHP analysis due to their demonstrated high sensitivity, selectivity, and relatively low cost[3,4]. DPV yields characteristic voltammograms exhibiting distinct peaks corresponding to the oxidation of individual DHPs, thereby facilitating their simultaneous determination within complex mixtures[1,5]. However, the analysis of DPV data, especially in cases involving multiple DHPs, can present significant challenges. Traditional analytical workflows often necessitate manual peak identification, baseline correction, and quantification. These processes are inherently subjective, time-consuming, and susceptible to error.

Current electrochemical data analysis workflows frequently rely on proprietary software solutions, which restrict portability and consequently hinder the development and dissemination of open-source alternatives[6]. To mitigate these limitations, we have developed a Jupyter Notebook designed to automate the analysis of DPV data obtained from the simultaneous detection of DHPs. Jupyter Notebooks provide a flexible and interactive computational environment conducive to data analysis, seamlessly integrating code execution, data visualization, and narrative documentation within a single, cohesive document. In this

regard, our notebook offers a user-friendly, modular interface that streamlines the entire analytical process, from initial data import through final visualization.

2. Materials and Methods

2.1 Chemicals

Catechol ($C_6H_6O_2$; Extrapure, SRL), hydroquinone ($C_6H_6O_2$; Extrapure, SRL), alizarin red S ($C_{14}H_7NaO_7S$; Avra Synthesis), potassium phosphate dibasic (K_2HPO_4; Extrapure, SRL), potassium phosphate monobasic (KH_2PO_4; Extrapure, SRL), and sulfuric acid (H_2SO_4; EMPARTA® ACS, Merck) were of analytical grade and used as received. Triple-distilled water was employed throughout all experiments. A 0.1 M phosphate buffer solution (PBS) was prepared from 1 M stock solutions of KH_2PO_4 and K_2HPO_4 and subsequently purged with argon for one hour before use. The pH of the 0.1 M PBS was adjusted to the desired value (±0.05) using either 5 M hydrochloric acid (HCl) or 5 M sodium hydroxide (NaOH). Working solutions of hydroquinone and catechol were prepared fresh from either 10 mM or 0.1 M stock solutions.

2.2 Electrochemical Measurements

Electrochemical measurements were performed using a Metrohm Autolab PGSTAT 302N potentiostat with a single-compartment cell, employing a modified Glassy Carbon Electrode GCE (working), platinum coil (counter), and SCE (reference) electrode. Measurements were conducted at 25 ± 1 °C. Voltammograms are background-subtracted unless noted. Poly(Alizarin Red S) modified GCEs (PARS/GCEs) were prepared as described elsewhere[7]. Hydroquinone (HQ) and catechol (CC) were determined simultaneously at the PARS/GCE in 0.1 M PBS (pH 7.0) using DPV (-0.1 to 0.4 V scan, 2 s quiet time). DPV parameters: 0.005 V increment, 0.05 V amplitude, 0.05 s pulse width, 0.5 s pulse period.

3. Software and Libraries

The Jupyter Notebook is compatible with Linux, macOS, and Windows systems running Python 3.7 or higher, provided that the dependencies listed in Table 1 are installed. For enhanced portability, the notebook can be readily deployed within a Google Colab workspace, obviating the need for local installations. Comprehensive usage instructions are available within the associated GitHub repository. Beyond the aforementioned software requirements, the notebook operates effectively on standard laptop or desktop computers. However, proper file naming conventions are essential for successful execution. Data files must be named according to the following format, corresponding to concentration units: xx_nM (nanomolar), xx_uM (micromolar), xx_mM (millimolar), and xx_M (molar). Furthermore, all data files must be placed within a directory named "data" for the notebook to function correctly.

Table.1. List of Python libraries used and their version dependencies

Library Name	jupyterlab	matplotlib	NumPy	pandas	scipy	seaborn	statsmodels
Version	>=4.1.4	>=3.8.3	>=1.26.4	>=2.2.1	>=1.12.0	>=0.13.2	>=0.14.1

3.5. Workflow of the data analysis in the Jupyter Notebook

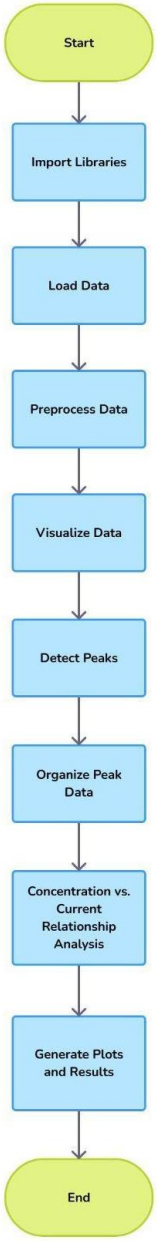

Fig.1. A flowchart describing the workflow in the Jupyter Notebook

3. Results and Discussion

The data loading and preprocessing stage began by importing essential Python libraries, including pandas for data manipulation, NumPy for numerical operations, matplotlib for basic plotting, and seaborn for enhanced visualizations. Data from text files were placed inside the folder named data and then loaded into Pandas DataFrames. Relevant columns, specifically potential and differential current, were extracted and renamed for improved clarity and ease of use in subsequent analysis.

For initial data visualization, the differential pulse voltammograms (DPV) for all concentrations were plotted using Seaborn's lineplot function (Fig.2). These plots revealed distinct peaks corresponding to the redox processes of the target analytes, HQ, and CC. A clear legend was generated, labeling each concentration for easy identification and interpretation of the plotted data.

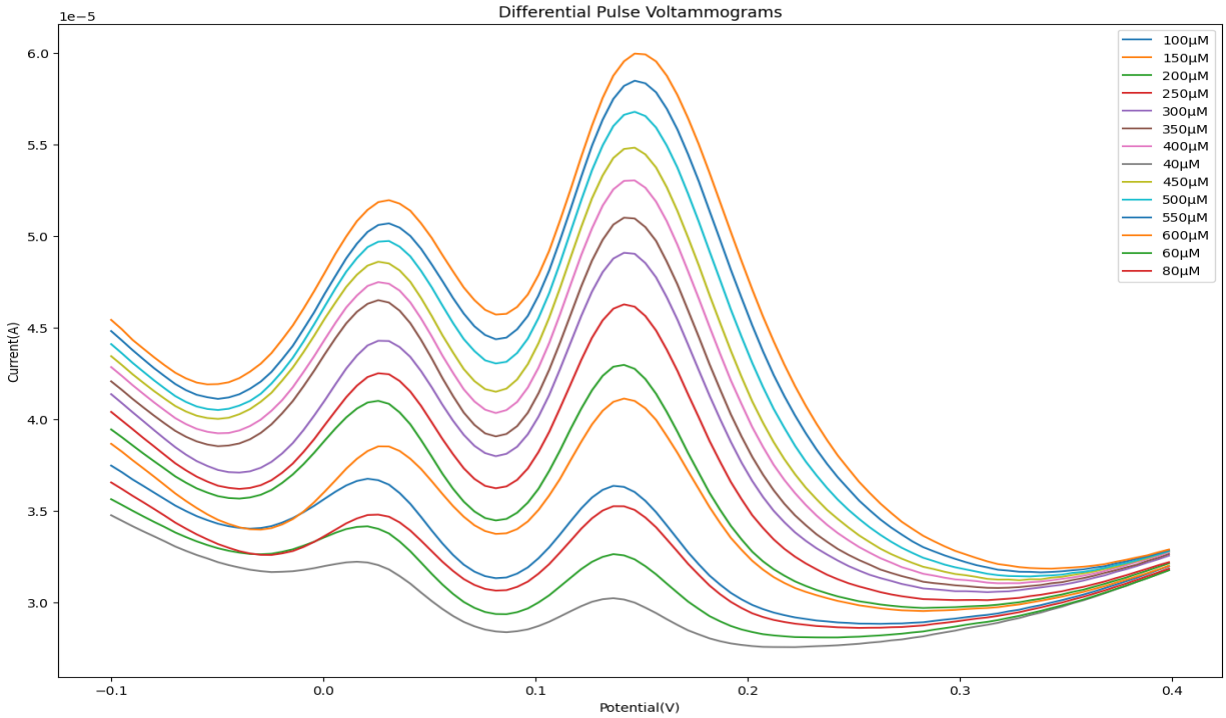

Fig.2. Differential Pulse Voltammograms plotted using matplotlib

Peak detection and analysis were performed by defining a custom function, find_peaks_and_values, designed to locate peaks within the current data and extract the corresponding potential and current values. SciPy's find_peaks function was employed to automatically detect these peaks, accurately identifying their potentials and currents. This function was applied iteratively to the data from each concentration. The extracted peak information was stored in a new pandas DataFrame named peak_df, to which a "Concentration(μM)" column was added to organize the data by concentration level (Fig.3).

	HQ_Peak_Potential(V)	CC_Peak_Potential(V)	HQ_Peak_Current(A)	CC_Peak_Current(A)	Concentration(µM)
40_mu_M	0.015869	0.136719	0.000032	0.000030	40.0
60_mu_M	0.020905	0.136719	0.000034	0.000033	60.0
80_mu_M	0.025940	0.136719	0.000035	0.000035	80.0
100_mu_M	0.020905	0.136719	0.000037	0.000036	100.0
150_mu_M	0.025940	0.141754	0.000039	0.000041	150.0
200_mu_M	0.025940	0.141754	0.000041	0.000043	200.0
250_mu_M	0.025940	0.141754	0.000043	0.000046	250.0
300_mu_M	0.025940	0.141754	0.000044	0.000049	300.0
350_mu_M	0.025940	0.141754	0.000046	0.000051	350.0
400_mu_M	0.025940	0.146790	0.000047	0.000053	400.0
450_mu_M	0.025940	0.146790	0.000049	0.000055	450.0
500_mu_M	0.030975	0.146790	0.000050	0.000057	500.0
550_mu_M	0.030975	0.146790	0.000051	0.000058	550.0
600_mu_M	0.030975	0.146790	0.000052	0.000060	600.0

Fig.3. Screenshot of the peak_df dataframe

To visually represent the identified peaks, the DPV data was plotted again, this time using Matplotlib's scatter function to highlight the locations of the detected peaks (Fig.4).

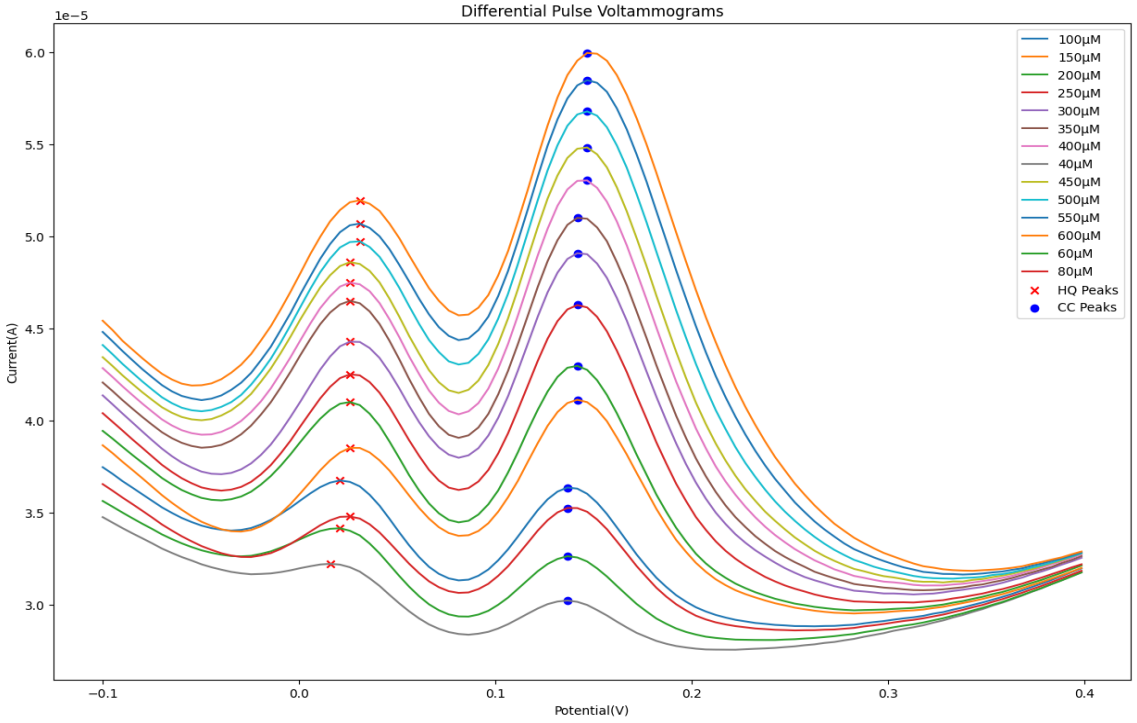

Fig.4. Differential Pulse Voltammograms plotted using Matplotlib along with identified peaks

For quantitative analysis, calibration curves were generated and linear regression was performed. The stats.linregress function was employed to calculate the linear relationship between concentration and peak current for both HQ and CC. This analysis revealed a strong linear correlation for both HQ and CC, indicating a proportional response between the concentration of the analyte and the magnitude of the electrochemical signal.

The calibration curves were then visualized using Seaborn's regplot function. The resulting plots included the calculated linear regression equation and the R-squared value within the legend, providing a comprehensive view of the calibration data and the goodness of fit for the linear model (Fig.5). High R-squared values, close to 1, indicated a good fit of the data to the linear model, supporting the validity of the concentration-current relationship.

The linear regression analysis yielded calibration curves for both HQ and CC, enabling the quantification of their concentrations in unknown samples. These calibration curves provided equations that could be used to determine the analyte concentration based on the measured peak current.

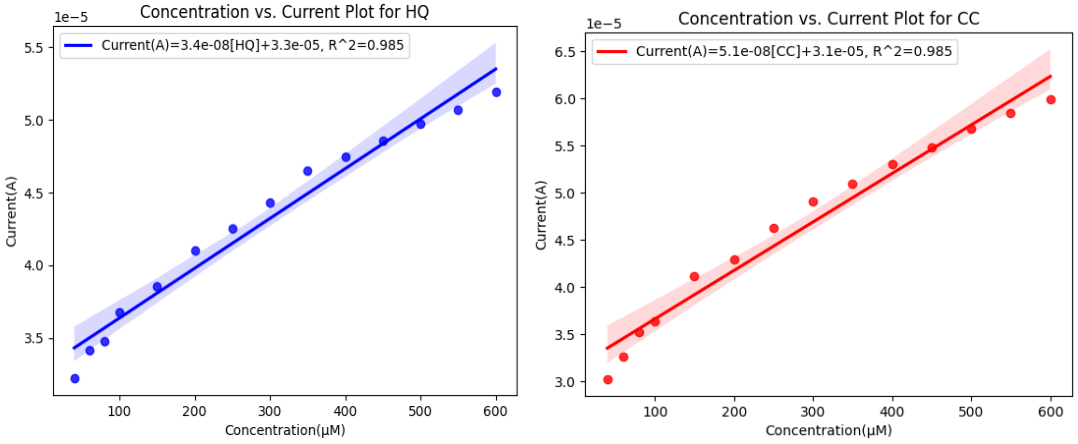

Fig.5. Linear calibration plots for HQ and CC and their corresponding linear fit equations

The sensitivity of the electrochemical method was evaluated by examining the slopes of the calibration curves. Steeper slopes indicated higher sensitivity, implying that smaller changes in concentration could be readily detected.

4. Conclusion

This article presents a Jupyter Notebook designed for the efficient and streamlined analysis of DPV data acquired from the simultaneous detection of dihydroxy phenols DHPs. By specifying the data directory, the entire analytical process can be executed with minimal user intervention. The notebook automates several key analytical procedures, thereby reducing manual effort and enhancing the accuracy and reproducibility of the analysis. This user-friendly tool empowers electrochemistry researchers in environmental monitoring and related fields to efficiently and accurately analyze DHP data, even with limited programming experience.

5. Future Studies and Limitations

The automated peak detection and analysis workflow implemented in this Jupyter Notebook provides a robust tool for electrochemical investigations. It streamlines the analytical process, improves accuracy, and facilitates the quantification of analytes across a range of applications. Future research may focus on integrating more sophisticated signal processing methodologies and expanding the notebook's functionality to encompass data from other electrochemical techniques and diverse instrumentation. It is acknowledged that this notebook is currently tailored for DPV data acquired using a Metrohm Autolab PGSTAT 302N potentiostat. Furthermore, the algorithm's accuracy requires comprehensive validation using data spanning a broader range of analyte concentrations.

Acknowledgments

The authors gratefully acknowledge Prof. R. Venkatesan of the Department of Chemistry, Pondicherry University, Pondicherry, India-605014, for granting access to and assistance with the instrumentation used in these experiments.

Author Contributions

A.N. conceived and designed the study, developed the code, and contributed to the manuscript. P.S. performed the experiments and led the manuscript preparation. All authors provided critical feedback on the experimental design, data analysis, data interpretation, and manuscript content.

Conflict of Interest

The authors declare that they have no conflict of interest.

Supplementary Material

The complete code, associated data, and further details are available in the project repository: https://github.com/anatarajank/Electrochemical-Sensor-Data-Analyzer. This repository is licensed under the Creative Commons Attribution 4.0 International License (CC-BY-4.0 International), which permits the free use, distribution, and adaptation of the work, provided the original author is properly credited.

References

1. H. Meskher and F. Achi, *Crit Rev Anal Chem*, **54**, 1354–1367 (2024) https://www.tandfonline.com/doi/abs/10.1080/10408347.2022.2114784.
2. L. H. Keith, *Compilation of Sampling Analysis Methods, US Environmental Protection Agency* L. H. Keith, Editor, First., Lewis Publishers, Boca Raton, FL, (1991).
3. P. S. Ganesh, S. E. Elugoke, S. H. Lee, S. Y. Kim, and E. E. Ebenso, *Inorg Chem Commun*, **172** (2025) https://www.sciencedirect.com/science/article/abs/pii/S1387700324017544.
4. M. Kaneva et al., *Microchemical Journal*, **204**, 111106 (2024) https://www.sciencedirect.com/science/article/abs/pii/S0026265X24012189.
5. C. Rong, Y. Huang, and X. Zheng, *Mater Chem Phys*, **329**, 130077 (2025) https://www.sciencedirect.com/science/article/abs/pii/S0254058424012057.

6. S. C. H. Lee and P. J. Burke, *Electrochim Acta*, **422**, 140481 (2022) https://www.sciencedirect.com/science/article/pii/S0013468622005618.
7. N. Aravindan and M. V. Sangaranarayanan, *Journal of Electroanalytical Chemistry*, **789**, 148–159 (2017) http://linkinghub.elsevier.com/retrieve/pii/S1572665717301315.