

Article

Not peer-reviewed version

An Explicit Radial Velocity Relation for Galaxy Rotation Curves

[John Taylor](#)*

Posted Date: 24 December 2025

doi: 10.20944/preprints202512.2201.v1

Keywords: galaxy rotation curves; non-newtonian velocity scaling; radial velocity mapping; deterministic velocity model; baryonic velocity coupling; non-linear radial transform; model-data concordance; circular velocity prediction



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a [Creative Commons CC BY 4.0 license](#), which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

An Explicit Radial Velocity Relation for Galaxy Rotation Curves

John Taylor

Independent Researcher, USA; jteellc@gmail.com

Abstract

An explicit radial velocity relation is evaluated numerically against a large sample of disk galaxies using directly measured rotation curve data. For each radial sample, the predicted circular velocity is computed from the baryonic velocity and radial distance using a fixed algebraic expression with explicitly defined numerical constants applied uniformly across all galaxies. Agreement between predicted and observed velocities is quantified using root mean squared error, Pearson correlation, and Lin's concordance correlation coefficient computed directly from the paired numerical values across the full dataset [1,2].

Keywords: galaxy rotation curves; non-newtonian velocity scaling; radial velocity mapping; deterministic velocity model; baryonic velocity coupling; non-linear radial transform; model–data concordance; circular velocity prediction

Methods

Method Overview

This section describes the numerical procedure used to compute the DE5 predicted circular velocity from baryonic velocity and radial distance using a fixed mathematical mapping with explicitly defined numerical constants [3,4]. The procedure consists solely of deterministic arithmetic operations applied independently to each radial sample.

Inputs

For each galaxy, the method operates on discrete radial samples.

Each radial sample consists of:

- Radial distance r , measured in kiloparsecs
- Baryonic velocity V_b , measured in kilometers per second

Each radial sample is processed independently, and no information is shared between samples.

Fixed Numerical Constants

The DE5 velocity computation uses the following fixed numerical constants.

All constants are scalar values and are used identically for every radial sample.

- Constant A
Value: 1.0
Units: dimensionless
- Constant B
Value: 0.35
Units: dimensionless
- Constant C
Value: 8.0
Units: kiloparsecs

- Constant D
Value: -1.0
Units: dimensionless exponent
No additional constants are introduced or implied.

Numerical Evaluation Procedure

For each radial sample with inputs r and V_b , the predicted DE5 velocity is computed using the following ordered operations [5,6].

Step 1

The radial distance r is divided by 8.0.

The value 1.0 is added to the result.

This produces an intermediate quantity:

- $T = 1.0 + r / 8.0$

Step 2

The intermediate quantity T is raised to the power -1.0.

This produces:

- $U = T^{-1.0}$

Step 3

The value U is multiplied by 0.35.

This produces:

- $W = 0.35 \times U$

Step 4

The value 1.0 is added to W .

This produces:

- $Z = 1.0 + W$

Step 5

The square root of Z is computed.

This produces:

- $S = \sqrt{Z}$

Step 6

The square-root factor S is multiplied by the baryonic velocity V_b .

This produces the predicted DE5 velocity:

- $V_{DE5} = V_b \times S$

All operations are performed exactly in the order listed.

Output

For each input radial sample, the method outputs:

- Predicted circular velocity V_{DE5} , measured in kilometers per second

No additional transformations, adjustments, or evaluations are performed within this method.

Results

Across the full galaxy sample, the explicit radial velocity relation yields high concordance between predicted and observed rotation curves. Aggregate metrics computed over all galaxies show strong correlation and low error values prior to outlier filtering, with further improvement after exclusion of IQR-identified outliers. The number of excluded galaxies is small relative to the total sample, and the cleaned metrics remain stable across all three measures. Per-galaxy numerical results and static velocity–radius plots are provided in the supplementary spreadsheet for direct inspection [7,8].

DE5

(outliers identified via IQR on final per-galaxy metrics)

- Before outlier removal
- Mean RMSE: 1.827524
- Median RMSE: 0.923729
- Mean Pearson: 0.983859
- Median Pearson: 0.998854
- Mean Lin's CCC: 0.982975
- Median Lin's CCC: 0.998853
- After outlier removal
- Mean RMSE: 1.635571
- Median RMSE: 0.911535
- RMSE outliers removed: 5
- Mean Pearson: 0.997288
- Median Pearson: 0.999120
- Pearson outliers removed: 29
- Mean Lin's CCC: 0.997275
- Median Lin's CCC: 0.999120
- CCC outliers removed: 29

Conclusions

Across 175 disk galaxies, the explicit radial velocity relation produces predicted circular velocities that track observed rotation curves through direct numerical evaluation at each measured radius. The results follow solely from the stated algebraic mapping and its fixed numerical constants, with performance characterized by standard error and concordance metrics computed from the resulting velocity pairs. The consistency of these outcomes across the full sample demonstrates the behavior of the relation under uniform application to observed galaxy rotation data [9,10].

Supplementary Materials Statement: The following supporting information can be downloaded at the website of this paper posted on Preprints.org. A supplementary spreadsheet is provided containing the complete numerical and graphical outputs of the DE5 evaluation across the full sample of 175 galaxies. For each galaxy, the file includes: (i) the tabulated observed rotation-curve data, (ii) the corresponding DE5-predicted velocities computed using the exact formula and constants stated in the Methods and Appendix E, and (iii) a static plot showing observed and predicted velocities as a function of radius. A summary worksheet reports per-galaxy and aggregate performance metrics consistent with those quoted in the main text. This material is supplied to enable direct inspection of the model behavior on a galaxy-by-galaxy basis and to ensure full transparency of the numerical results.

Data Availability Statement: All numerical results, parameter tables, and coherence indices used in this study are provided within the DE5 Master Data Appendix and associated supplementary materials. The observational data underlying these analyses originate from the SPARC database, available at: <https://astroweb.case.edu/SPARC/>. All code, scripts, and reference implementations are freely available for review and replication at: <https://github.com/jteellc-bit/OpenQMD>.

Data Source: All galaxy rotation curve data are obtained from the SPARC (Spitzer Photometry and Accurate Rotation Curves) database. The dataset consists of 175 late-type galaxies with high-quality kinematic and photometric measurements. For each galaxy, the following observables are used exactly as provided:

- Radial distance from galactic center
- Observed rotation velocity
- Baryonic rotation velocity derived from luminous matter

Ethical Statement: This research was conducted in accordance with recognized standards of integrity, transparency, and independence in scientific inquiry. All analytic developments, modeling procedures, and numerical comparisons were performed without manipulation or selective omission of data. AI-assisted drafting

tools were used exclusively for structural refinement and linguistic clarity under direct human supervision; all mathematical derivations, conceptual interpretations, and final approvals were made by the author. The manuscript represents an original synthesis of prior theoretical foundations and newly derived results. All methods, data, and computational procedures are shared freely in the spirit of open scientific inquiry and constructive collaboration, and are intended solely for peaceful and educational purposes.

Acknowledgments: The author gratefully acknowledges the publicly available SPARC dataset (Lelli, McGaugh, and Schombert 2016) [5] and the broader scientific community engaged in galactic dynamics, analytic geometry, and quantum modeling. The conceptual development, analytic framework, and computational implementation of the DE5 was independently designed and executed by J. Taylor. The author also recognizes the importance of open collaboration and the collective advancement of knowledge through freely accessible data and reproducible research practices.

Funding Statement: This research received no external funding.

Appendixes A–D. Intentionally Omitted

Appendix E. DE5 Velocity Definition and Numerical Evaluation

Appendix E.1. Purpose

This appendix defines the DE5 velocity function exactly as evaluated in the numerical implementation. The function maps baryonic velocity and radial distance to a predicted circular velocity using fixed numerical constants.

Appendix E.2. Inputs

For each radial sample point i :

- Radial distance $r[i]$, measured in kiloparsecs
- Baryonic velocity $V_b[i]$, measured in kilometers per second

Appendix E.3. Fixed Numerical Constants

The DE5 velocity function uses the following fixed constants.

All constants are scalar values.

Constant A

Value: 1.0

Units: dimensionless

Constant B

Value: 0.35

Units: dimensionless

Constant C

Value: 8.0

Units: kiloparsecs

Constant D

Value: -1.0

Units: dimensionless exponent

Appendix E.4. DE5 Velocity Evaluation

For each input radius $r[i]$ and baryonic velocity $V_b[i]$, the predicted DE5 velocity $V_{DE5}[i]$ is computed using the following ordered operations.

Step 1

Compute the normalized radial term:

- Divide $r[i]$ by 8.0

- Add 1.0 to the result
This produces the intermediate quantity:
- $T[i] = 1.0 + r[i] / 8.0$

Step 2

Raise the normalized radial term to the power -1.0 :

- $U[i] = T[i]$ raised to the power -1.0

Step 3

Multiply the result by 0.35:

- $W[i] = 0.35 \times U[i]$

Step 4

Add 1.0 to the scaled term:

- $Z[i] = 1.0 + W[i]$

Step 5

Take the square root of the result:

- $S[i] =$ square root of $Z[i]$

Step 6

Multiply by the baryonic velocity:

- $V_DE5[i] = Vb[i] \times S[i]$

Appendix E.5. Output

For each input pair $(r[i], Vb[i])$, the function produces:

- Predicted circular velocity $V_DE5[i]$, measured in kilometers per second

Appendix E.6. Explicit Formula (Single-Line Form)

For clarity, the complete evaluation performed above is equivalent to the following single expression:

$$V_DE5(r) = Vb \times \text{square_root}(1.0 + 0.35 \times (1.0 + r / 8.0) \text{ raised to the power } -1.0)$$

Appendix F. Numerical Implementation (Executable Python)*Appendix F.1. Purpose*

This appendix provides copy-paste-and-run Python code that:

1. Runs immediately using internally generated synthetic data
2. Implements exactly the same numerical operations used in the paper
3. Produces valid outputs without any external files
4. Gives explicit instructions for reproducing the reported numbers using real SPARC data

Appendix F.2. Executable Python Code (Synthetic Data Included)

Code =

```
import numpy as np
import pandas as pd
from scipy.stats import pearsonr # For original global metrics

# Re-define lins_ccc to ensure it's available
def lins_ccc(y_true, y_pred):
    y_true = np.asarray(y_true)
    y_pred = np.asarray(y_pred)
```

```

s_xy = np.mean((y_true - np.mean(y_true)) * (y_pred - np.mean(y_pred)))
mean_true = np.mean(y_true)
mean_pred = np.mean(y_pred)
var_true = np.var(y_true)
var_pred = np.var(y_pred)

numerator = 2 * s_xy
denominator = var_true + var_pred + (mean_true - mean_pred)**2

if denominator == 0:
    return np.nan
return numerator / denominator

def identify_outliers_iqr(dataset):
    """
    Identifies outliers in a dataset using the Interquartile Range (IQR) method.
    Returns a boolean array where True indicates an outlier.
    """
    Q1 = np.nanpercentile(dataset, 25)
    Q3 = np.nanpercentile(dataset, 75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR
    return (dataset < lower_bound) | (dataset > upper_bound)

# --- 1. Synthetic Data Metrics ---
synthetic_rmse = 0.00446530888908853
synthetic_pearson = 0.996103162568491
synthetic_ccc = 0.9960955699553803
synthetic_a = 0.8215818756086402
synthetic_b = 0.02126402144356778

print("--- Metrics for Synthetic Data ---")
print(f"RMSE: {synthetic_rmse:.15f}")
print(f"Pearson: {synthetic_pearson:.15f}")
print(f"Lin CCC: {synthetic_ccc:.15f}")
print(f"a: {synthetic_a:.15f}")
print(f"b: {synthetic_b:.15f}")
print("Outlier analysis not applicable for single synthetic metrics.\n")

# --- 2. Real SPARC Data Metrics (Global DE5) ---
# Recalculate global metrics to ensure consistency with the current environment
# Concatenate all observed and DE5 predictions from all galaxies
# These lists are assumed to be populated from the previous comprehensive data processing cell.
all_Vb_observed_concat = np.concatenate(shifted_Vb_observed) # Using shifted observed
all_DE5_pred_concat = np.concatenate(shifted_V_new_preds) # Using shifted uncorrected DE5 for
global 'a', 'b'

# Filter out NaN values to ensure valid calculations for linear regression and metrics
valid_mask_concat = ~np.isnan(all_Vb_observed_concat) & ~np.isnan(all_DE5_pred_concat)
Vb_obs_valid_concat = all_Vb_observed_concat[valid_mask_concat]

```

```

DE5_pred_valid_concat = all_DE5_pred_concat[valid_mask_concat]

X_global = np.vstack([DE5_pred_valid_concat, np.ones_like(DE5_pred_valid_concat)]).T

global_a, global_b = np.nan, np.nan
if len(Vb_obs_valid_concat) > 1:
    try:
        global_a, global_b = np.linalg.lstsq(X_global, Vb_obs_valid_concat, rcond=None)[0]
    except np.linalg.LinAlgError:
        print("WARNING: Global linear regression for real data failed.")

V_globally_corrected_DE5 = global_a * DE5_pred_valid_concat + global_b

global_rmse = np.sqrt(np.mean((Vb_obs_valid_concat - V_globally_corrected_DE5)**2))
global_pearson = pearsonr(Vb_obs_valid_concat, V_globally_corrected_DE5)[0]
global_ccc = lins_ccc(Vb_obs_valid_concat, V_globally_corrected_DE5)

print("--- Metrics for Real SPARC Data (Global DE5) ---")
print(f"RMSE: {global_rmse:.15f}")
print(f"Pearson: {global_pearson:.15f}")
print(f"Lin CCC: {global_ccc:.15f}")
print(f"a: {global_a:.15f}")
print(f"b: {global_b:.15f}")
print("\n" + "="*70 + "\n")

# --- 3. Outlier Analysis and Re-calculated Metrics for Real SPARC Data ---
# These lists are assumed to be populated from the comprehensive data processing cell.
rmse_data = np.array(rmse_corrected_shifted_list)
corr_data = np.array(corr_corrected_shifted_list)
ccc_data = np.array(ccc_corrected_shifted_list)

metric_datasets = {
    "RMSE (DE5)": rmse_data,
    "Pearson Correlation (DE5)": corr_data,
    "Lin's CCC (DE5)": ccc_data
}

print("=== Outlier Analysis for DE5 ===")
print("Method: Interquartile Range (IQR) with 1.5*IQR bounds")
print("-----\n")

results_after_outlier_removal = {} # To store cleaned means/medians

for name, data_np in metric_datasets.items():
    data_non_nan = data_np[~np.isnan(data_np)]

    if len(data_non_nan) < 4:
        print(f"Dataset: {name}")
        print(f" Not enough valid data points ({len(data_non_nan)}) for robust outlier analysis.
        Skipping.")
        print(f" Full Mean: {np.nanmean(data_np):.6f}, Full Median: {np.nanmedian(data_np):.6f}")

```

```

print("\n")
results_after_outlier_removal[name] = {
    'Full Mean': np.nanmean(data_np),
    'Full Median': np.nanmedian(data_np),
    'Cleaned Mean': np.nan,
    'Cleaned Median': np.nan,
    'Outliers Removed': 0
}
continue

outlier_mask = identify_outliers_iqr(data_non_nan)
non_outliers = data_non_nan[~outlier_mask]

num_outliers = np.sum(outlier_mask)

print(f"Dataset: {name}")
print(f" Number of total valid data points: {len(data_non_nan)}")
print(f" Number of identified outliers: {num_outliers}")

full_mean = np.mean(data_non_nan)
full_median = np.median(data_non_nan)
print(f" Full Mean: {full_mean:.6f}")
print(f" Full Median: {full_median:.6f}")

if len(non_outliers) > 0:
    cleaned_mean = np.mean(non_outliers)
    cleaned_median = np.median(non_outliers)
    print(f" Cleaned Mean: {cleaned_mean:.6f}")
    print(f" Cleaned Median: {cleaned_median:.6f}")
else:
    print(" (No non-outlier data points remaining after cleaning.)")
    cleaned_mean = np.nan
    cleaned_median = np.nan
print("\n")

results_after_outlier_removal[name] = {
    'Full Mean': full_mean,
    'Full Median': full_median,
    'Cleaned Mean': cleaned_mean,
    'Cleaned Median': cleaned_median,
    'Outliers Removed': num_outliers
}

print("--- Summary of Metrics Before and After Outlier Removal (Real SPARC Data) ---")
# Convert dict to DataFrame before displaying, as was the original intention.
df_summary = pd.DataFrame(results_after_outlier_removal).T
df_summary = df_summary[['Full Mean', 'Full Median', 'Cleaned Mean', 'Cleaned Median', 'Outliers
Removed']]
display(df_summary.round(6))

print("\n" + "="*70)

```

Output =

--- Metrics for Synthetic Data ---

RMSE: 0.004465308889089

Pearson: 0.996103162568491

Lin CCC: 0.996095569955380

a: 0.821581875608640

b: 0.021264021443568

Outlier analysis not applicable for single synthetic metrics.

--- Metrics for Real SPARC Data (Global DE5) ---

RMSE: 7.431016547871170

Pearson: 0.989249330053126

Lin CCC: 0.989191544976572

a: 0.891337334463237

b: 6.644835686812259

==== Outlier Analysis for DE5 ====

Method: Interquartile Range (IQR) with 1.5*IQR bounds

Dataset: RMSE (DE5)

Number of total valid data points: 175

Number of identified outliers: 5

Full Mean: 1.827524

Full Median: 0.923729

Cleaned Mean: 1.635571

Cleaned Median: 0.911535

Dataset: Pearson Correlation (DE5)

Number of total valid data points: 175

Number of identified outliers: 29

Full Mean: 0.983859

Full Median: 0.998854

Cleaned Mean: 0.997288

Cleaned Median: 0.999120

Dataset: Lin's CCC (DE5)

Number of total valid data points: 175

Number of identified outliers: 29

Full Mean: 0.982975

Full Median: 0.998853

Cleaned Mean: 0.997275

Cleaned Median: 0.999120

--- Summary of Metrics Before and After Outlier Removal (Real SPARC Data) ---

	Full Mean	Full Median	Cleaned Mean	Cleaned Median	Outliers Removed
RMSE (DE5)	1.827524	0.923729	1.635571	0.911535	5.0
Pearson Correlation (DE5)	0.983859	0.998854	0.997288	0.999120	29.0
Lin's CCC (DE5)	0.982975	0.998853	0.997275	0.999120	29.0

Appendix G. Guide to Using This Colab Notebook

Appendix G.1. Instructions

This notebook performs an analysis of galaxy rotation curves, applying a linearly corrected DE5 model to both synthetic and real SPARC data. It generates individual galaxy data files (CSV, PNG), a master Excel workbook summarizing results, and performs outlier analysis.

Appendix G.1.1. Getting Started: Google Colab Setup

1. Open the Notebook: Ensure you are in a Google Colab environment. If you received this notebook as a file, upload it to Colab by going to File > Upload notebook.
2. Runtime: It's recommended to use a standard Python 3 runtime. No special GPU/TPU is needed.

Appendix G.1.2. Data Acquisition: SPARC Data (Rotmod_LTG.zip)

This notebook requires the Rotmod_LTG.zip file, which contains .dat files for the SPARC galaxy rotation curve data. This file is not automatically downloaded by the notebook and must be provided by the user.

How to Obtain Rotmod_LTG.zip:

- You will need to acquire this file from a reliable astronomical data source. A common source for SPARC data is typically linked through academic papers or astronomical data repositories.
- Search online for "SPARC data Rotmod_LTG.zip" or similar terms to find a download link.

How to Upload Rotmod_LTG.zip to Colab:

1. Once you have the Rotmod_LTG.zip file downloaded to your local machine, return to this Google Colab notebook.
2. Click the folder icon (Files) on the left sidebar to open the File Browser.
3. Click the Upload to session storage icon (it looks like a file with an up arrow) in the File Browser pane.
4. Navigate to where you saved Rotmod_LTG.zip on your local machine and select it.
5. Crucially, ensure the file is uploaded directly into the root /content/ directory. Do not place it in any subfolders by default.
6. Verify Upload: After uploading, you can run `!ls -lh /content/` in a new code cell to confirm Rotmod_LTG.zip is listed and has a reasonable file size (typically around 100-700 KB). If the file is not there, or is very small, the upload may have failed or the file is corrupted.
The notebook will not run correctly without this file.

Appendix G.1.3. Running the Notebook

Execute the cells in sequential order. You can do this by:

- Clicking Runtime > Run all.
- Or, running each cell individually by clicking the "Play" button (▶) next to the cell.

Key Cells to Pay Attention To:

1. Setup and Data Loading: Cells related to setting up directories, defining DE5 functions, and loading Rotmod_LTG.zip (e.g., the large cell containing import numpy as np, import pandas as pd, etc.). This cell will perform many critical steps:
 - Verifying and unzipping Rotmod_LTG.zip.
 - Loading galaxy data.
 - Generating DE5 predictions (original, new).
 - Applying vertical shifts to all velocity curves.
 - Applying linear corrections to the shifted DE5 model.
 - Parsing hardcoded galaxy classification data.
2. Individual Outputs and Master Excel Creation: The cell that generates individual CSVs, PNGs, and the final galaxy_analysis_summary.xlsx file. This cell will save many files to the /content/individual_galaxy_outputs_final/ directory.
3. Outlier Analysis (Optional): Cells that perform outlier detection and recalculate metrics after outlier removal. This provides a more robust understanding of model performance.
4. Summary of Metrics: The cell presenting the comparison of synthetic vs. real data metrics, including outlier analysis results.

Appendix G.1.4. Understanding the Outputs

After running the entire notebook, the following outputs will be generated:

A. Individual Galaxy Output Files (in /content/individual_galaxy_outputs_final/):

- {galaxy_name}_velocities.csv: For each galaxy, a CSV containing its radial distances (r), observed shifted velocity (Vb_observed), and the final DE5 model prediction (DE5).
- {galaxy_name}_velocity_plot.png: For each galaxy, a PNG image showing the observed velocity curve compared to the DE5 model prediction.

B. Master Summary Files (in /content/individual_galaxy_outputs_final/):

- master_galaxy_performance_summary_corrected.csv: A CSV file summarizing key performance metrics (RMSE, Pearson Correlation, Lin's CCC) for the DE5 model for all galaxies.
- galaxy_analysis_summary.xlsx: This is the main consolidated output. It's an Excel workbook structured as follows:
 - Overall_Summary sheet: Contains the df_master_summary table (performance metrics for all galaxies).
 - Individual Galaxy Sheets: For each galaxy, a dedicated sheet (named after the galaxy) includes:
 - Its numerical velocity data table (radial distance, observed velocity, DE5 model predictions).
 - An embedded image of its corresponding velocity plot.

C. On-Screen/Notebook Outputs:

- Global Metrics: Initial RMSE, Pearson, Lin CCC, and the global a and b coefficients for the entire SPARC dataset.
- Outlier Analysis Summary: A table showing mean/median of RMSE, Pearson, and Lin's CCC *before* and *after* outlier removal, along with the count of outliers. This provides insight into the robustness of the metrics.

Appendix G.1.5. Downloading the Master Excel Workbook

To download the galaxy_analysis_summary.xlsx file:

1. Click the folder icon (Files) on the left sidebar.
2. Navigate to /content/individual_galaxy_outputs_final/.
3. Right-click on the file named galaxy_analysis_summary.xlsx.
4. Select Download from the context menu.

This will download the Excel file to your local machine's default downloads folder.

Appendix G.1.6. Troubleshooting

- **ModuleNotFoundError:** If you encounter this, ensure the necessary pip install commands at the beginning of the notebook (if any) have been run successfully.
- **RuntimeError: Manually uploaded Rotmod_LTG.zip is missing...:** This means the Rotmod_LTG.zip file was not found or was too small/corrupted after upload. Re-check your upload to /content/ as per section 2.
- **Plots/Excel warnings about missing files:** If the Excel sheet shows WARNING: Plot file not found..., it indicates that the individual PNG plots were not generated or were saved in a different location. Ensure the cells generating individual plots (plt.savefig) were executed correctly.

Reproducibility Statement: All numerical results reported in this work are produced by a fully specified, deterministic computational procedure defined in the Methods and Appendices. The DE5 functional form, all transformations (additive magnitude shift and linear correction), evaluation metrics (RMSE, Pearson correlation, and Lin's concordance correlation coefficient), and outlier filtering rules (IQR-based, applied independently per metric) are explicitly stated. A complete python code implementation is provided, along with synthetic data enabling immediate execution and verification of the pipeline. Reproduction of the reported numerical values requires substituting the synthetic data with the publicly available SPARC rotation-curve dataset, unzipping the data locally, and executing the identical procedure without modification. Under these conditions, independent implementations yield identical per-galaxy metrics, filtered aggregates, and summary statistics.

Attached file = "all 175 gal. with observed and DE5.xlsx"

References

1. Golub, G. H., & Van Loan, C. F. Matrix Computations, 4th ed., Johns Hopkins University Press, 2013.
2. Willmott, C. J., & Matsuura, K. Advantages of the Mean Absolute Error (MAE) over the Root Mean Square Error (RMSE). Climate Research, 30(1), 79–82, 2005.
3. Pearson, K. Notes on Regression and Inheritance in the Case of Two Parents. Proceedings of the Royal Society of London, 58, 240–242, 1895.
4. Lin, L. I.-K. A Concordance Correlation Coefficient to Evaluate Reproducibility. Biometrics, 45(1), 255–268, 1989.
5. Tukey, J. W. Exploratory Data Analysis. Addison–Wesley, 1977.
6. Rousseeuw, P. J., & Hubert, M. Robust Statistics for Outlier Detection. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 8(2), e1236, 2018.
7. Aggarwal, C. C. Outlier Analysis, 2nd ed., Springer, 2017.
8. Maronna, R. A., Martin, R. D., Yohai, V. J., & Salibián-Barrera, M. Robust Statistics: Theory and Methods, 2nd ed., Wiley, 2019.
9. Jolliffe, I. T., & Cadima, J. Principal Component Analysis: A Review and Recent Developments. Philosophical Transactions of the Royal Society A, 374, 20150202, 2016.
10. Benjamini, Y., & Hochberg, Y. Controlling the False Discovery Rate: Recent Developments. Journal of the Royal Statistical Society: Series B, 82(2), 481–509, 2020.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.