

Walmart Sales Forecasting

Adil Ibraheem Koyava

30. Juni 2025

- 1 Introduction
- 2 Literature Review Article: 1
- 3 Literature Review Article: 2
- 4 Literature Review Article: 3
- 5 Literature Review Article: 4
- 6 Literature Review Article: 5
- 7 Literature Review Article: 6
- 8 Literature Review Article: 7

- 9 Literature Review Article: 8
- 10 Literature Review Article: 9
- 11 Literature Review Article: 10
- 12 Literature Review Article: 11
- 13 Literature Review Article: 12
- 14 Literature Review Article: 13
- 15 Literature Review Article: 14
- 16 Literature Review Article: 15

17 Literature Review Article: 16

18 Literature Review Article: 17

19 Literature Review Article: 18

20 Literature Review Article: 19

21 Literature Review Article: 20

22 Summary and Conclusion

23 Quellen

Introduction

Time series analysis and forecasting have become fundamental components of modern data science, with applications spanning from retail sales prediction to financial market analysis. The integration of powerful Python libraries such as pandas, NumPy, statsmodels, and specialized forecasting packages like pmdarima has revolutionized how we approach temporal data analysis.

This literature review examines the current state of time series forecasting methodologies, focusing on ARIMA modeling, exponential smoothing techniques, and the Python ecosystem tools that enable efficient implementation and deployment of forecasting solutions. We explore both theoretical foundations and practical applications, with particular attention to retail forecasting scenarios and modern deployment frameworks.

Literature Review Article: 1

Time Series Analysis: Forecasting and Control I

Authors: Box, George E. P.; Jenkins, Gwilym M.; Reinsel, Greta C.; Ljung, Gregory M.

Description: This definitive resource on Box-Jenkins methodology covers ARIMA modeling, transfer functions, intervention analysis, and multivariate time series. The fifth edition includes updated material on state-space models and nonlinear time series, providing comprehensive coverage of classical time series analysis techniques [Box+16].

Keywords: ARIMA, Box-Jenkins methodology, Transfer functions, Intervention analysis, State-space models

Disadvantages: The mathematical rigor may be challenging for practitioners without strong statistical backgrounds. The focus on classical methods may not adequately address modern machine learning approaches to time series analysis.

Conclusion: This foundational work remains essential for understanding the theoretical underpinnings of ARIMA modeling and provides the

Time Series Analysis: Forecasting and Control II

mathematical framework that underlies modern automated forecasting tools like pmdarima.

Literature Review Article: 2

Forecasting: Principles and Practice I

Authors: Hyndman, Rob J. and Athanasopoulos, George

Description:

This comprehensive textbook provides both theoretical foundations and practical applications of forecasting methods. The third edition covers modern forecasting techniques with extensive R examples, offering accessible explanations of complex concepts including ARIMA modeling, exponential smoothing, and forecast evaluation metrics [HA21].

Keywords:

Forecasting principles, Exponential smoothing, ARIMA models, Forecast evaluation, Time series decomposition

Disadvantages:

The primary focus on R programming may limit direct applicability for Python-centric workflows, though the theoretical concepts remain universally applicable.

Conclusion:

Forecasting: Principles and Practice II

This resource provides essential theoretical knowledge for understanding forecasting principles and serves as a foundation for implementing automated forecasting systems using Python libraries like pmdarima.

Literature Review Article: 3

Automatic Time Series Forecasting: The forecast Package for R I

Authors: Hyndman, Rob J. and Khandakar, Yeasmin

Description:

This foundational paper introduces the Auto ARIMA algorithm and its implementation in the forecast package. It describes the stepwise search procedure and statistical tests used for automatic model selection, providing the theoretical basis for automated ARIMA modeling that has been adapted in Python through pmdarima [HK08].

Keywords:

Auto ARIMA, Automatic model selection, Stepwise search, Information criteria, Model diagnostics

Disadvantages:

The algorithm may be computationally intensive for large datasets and might not always select the optimal model for specific domain applications without manual intervention.

Conclusion:

Automatic Time Series Forecasting: The forecast Package for R II

This paper establishes the foundation for automated ARIMA modeling and directly influences the implementation of modern Python forecasting libraries, making complex time series analysis accessible to practitioners.

Literature Review Article: 4

Exponential Smoothing: The State of the Art—Part II I

Authors: Gardner, Everette S. Jr.

Description:

This comprehensive review examines exponential smoothing methods, covering theoretical developments, practical applications, and comparative performance analysis. The paper provides a modern perspective on the evolution and effectiveness of exponential smoothing in contemporary forecasting practice [Jr.06].

Keywords:

Exponential smoothing, Holt-Winters method, Seasonal forecasting, Smoothing parameters, Forecast accuracy

Disadvantages:

The review predates many modern developments in machine learning and ensemble methods that have enhanced traditional exponential smoothing approaches.

Conclusion:

Exponential Smoothing: The State of the Art—Part II II

This work provides essential background for understanding exponential smoothing techniques and their continued relevance in modern forecasting applications, particularly when implemented through Python libraries.

Literature Review Article: 5

Forecasting Seasonals and Trends by Exponentially Weighted Moving Averages I

Authors: Winters, Peter R.

Description:

This foundational paper introduces the Holt-Winters exponential smoothing method for forecasting time series with trend and seasonal components. It establishes the mathematical framework for additive and multiplicative seasonal models that remain standard in modern forecasting practice [Win60].

Keywords:

Holt-Winters method, Seasonal forecasting, Exponential smoothing, Trend analysis, Multiplicative seasonality

Disadvantages:

The original formulation lacks the robustness and automation features found in modern implementations, requiring manual parameter tuning and model specification.

Conclusion:

Forecasting Seasonals and Trends by Exponentially Weighted Moving Averages II

This seminal work establishes the theoretical foundation for seasonal forecasting methods that continue to be widely used and have been enhanced through modern Python implementations in libraries like statsmodels and pmdarima.

Literature Review Article: 6

Array programming with NumPy I

Authors: Harris, Charles R. and Millman, K. Jarrod and others

Description:

This seminal research paper, published in Nature, describes NumPy's architecture, design principles, and impact on scientific computing. It demonstrates NumPy's significance as the foundation of the Python scientific computing ecosystem and its role in enabling efficient numerical operations for time series analysis [Har+20].

Keywords:

NumPy, Scientific computing, Array programming, Python ecosystem, Numerical computing

Disadvantages:

As a foundational library, NumPy requires additional specialized packages for specific time series operations and lacks built-in forecasting capabilities.

Conclusion:

Array programming with NumPy II

NumPy serves as the computational foundation for all time series analysis in Python, enabling efficient data manipulation and mathematical operations essential for forecasting applications.

Literature Review Article: 7

Data Structures for Statistical Computing in Python I

Authors: McKinney, Wes

Description:

This foundational paper introduces pandas data structures and design philosophy, establishing the framework for efficient data manipulation in Python. It describes the DataFrame and Series objects that have become essential for time series analysis and data preparation in forecasting workflows [McK10].

Keywords:

Pandas, DataFrame, Data structures, Data manipulation, Time series indexing

Disadvantages:

While excellent for data manipulation, pandas requires integration with specialized libraries for advanced statistical modeling and forecasting operations.

Conclusion:

Data Structures for Statistical Computing in Python II

Pandas provides the essential data manipulation capabilities that enable efficient preprocessing, analysis, and visualization of time series data in modern forecasting pipelines.

Literature Review Article: 8

Statsmodels: Econometric and Statistical Modeling with Python I

Authors: Seabold, Skipper and Perktold, Josef

Description:

This paper introduces statsmodels architecture and design principles, establishing a comprehensive statistical modeling library for Python. It provides the foundation for implementing classical econometric and statistical methods, including time series analysis and ARIMA modeling [SP10].

Keywords:

Statistical modeling, Econometrics, Time series analysis, Statistical inference, Python implementation

Disadvantages:

The library's focus on statistical rigor can result in more complex APIs compared to simplified forecasting packages, requiring deeper statistical knowledge for effective use.

Conclusion:

Statsmodels: Econometric and Statistical Modeling with Python II

Statsmodels provides the statistical foundation for rigorous time series analysis in Python, offering comprehensive tools for model estimation, diagnostics, and inference that complement automated forecasting approaches.

Literature Review Article: 9

Python for Data Analysis: Data Wrangling with pandas, NumPy, and Jupyter I

Authors: McKinney, Wes

Description:

This authoritative guide covers the Python data analysis ecosystem, including visualization best practices and integration patterns with various libraries. The third edition provides comprehensive coverage of pandas functionality and demonstrates effective workflows for data analysis and time series processing [McK23].

Keywords:

Data analysis, Pandas workflows, Data visualization, Time series processing, Python ecosystem

Disadvantages:

While comprehensive for general data analysis, the book provides limited coverage of specialized forecasting techniques and automated model selection procedures.

Conclusion:

Python for Data Analysis: Data Wrangling with pandas, NumPy, and Jupyter II

This resource provides essential practical knowledge for implementing effective data analysis workflows that support time series forecasting applications using the Python ecosystem.

Literature Review Article: 10

Matplotlib: A 2D Graphics Environment I

Authors: Hunter, John D.

Description:

This seminal paper introduces Matplotlib and its design philosophy, describing the library's architecture and demonstrating its capabilities for scientific computing visualization. It establishes the foundation for data visualization in Python that supports time series analysis and forecasting result presentation [Hun07].

Keywords:

Data visualization, Scientific plotting, Time series visualization, Python graphics, Statistical graphics

Disadvantages:

The original design predates modern interactive visualization needs, requiring additional libraries for sophisticated interactive dashboards and web-based applications.

Conclusion:

Matplotlib: A 2D Graphics Environment II

Matplotlib provides the fundamental visualization capabilities essential for exploratory time series analysis, model diagnostics, and forecast result presentation in Python-based forecasting workflows.

Literature Review Article: 11

Joblib: Optimizing Python for Scientific Computing I

Authors: Varoquaux, Gael; Gramfort, Alexandre; Pedregosa, Fabian

Description:

This academic paper details Joblib's design principles, performance optimizations for NumPy arrays, and applications in scientific computing workflows. It provides theoretical foundation for caching mechanisms and parallel processing strategies that enhance computational efficiency in time series analysis [VGP22].

Keywords:

Parallel processing, Memory caching, Performance optimization, Scientific computing, Computational efficiency

Disadvantages:

The focus on general scientific computing may not address specific optimization needs for time series forecasting workflows and specialized model training procedures.

Conclusion:

Joblib: Optimizing Python for Scientific Computing II

Joblib provides essential computational optimization capabilities that enable efficient processing of large-scale time series datasets and accelerate model training in forecasting applications.

Literature Review Article: 12

seaborn: statistical data visualization I

Authors: Waskom, Michael L.

Description:

This peer-reviewed paper describes seaborn's design philosophy, statistical capabilities, and integration with the scientific Python ecosystem. It establishes seaborn as a high-level interface for creating informative statistical visualizations that support time series analysis and forecasting result interpretation [Was21].

Keywords:

Statistical visualization, Exploratory data analysis, Time series plotting, Distribution analysis, Correlation visualization

Disadvantages:

While excellent for statistical visualization, seaborn lacks specialized time series plotting capabilities and forecasting-specific visualization functions.

Conclusion:

seaborn: statistical data visualization II

Seaborn enhances the visualization capabilities of the Python ecosystem, providing sophisticated statistical graphics that support exploratory time series analysis and model result communication.

Literature Review Article: 13

Retail forecasting: Research and practice I

Authors: Fildes, Robert and Goodwin, Paul

Description:

This comprehensive review examines retail forecasting challenges, methodologies, and practical considerations. It provides insights into the specific requirements and constraints of retail time series forecasting, including demand patterns, promotional effects, and inventory management considerations [FG19].

Keywords:

Retail forecasting, Demand prediction, Promotional effects, Inventory management, Seasonal patterns

Disadvantages:

The focus on retail applications may limit generalizability to other time series forecasting domains, and the review may not cover the latest developments in machine learning approaches.

Conclusion:

Retail forecasting: Research and practice II

This work provides valuable domain-specific insights for retail forecasting applications and demonstrates the practical considerations essential for implementing effective forecasting systems in commercial environments.

Literature Review Article: 14

Interactive Web-Based Data Visualization with R, plotly, and shiny I

Authors: Sievert, Carson

Description:

This comprehensive guide covers interactive data visualization principles and implementation, with extensive coverage of Plotly's architecture and capabilities. While focused on R, the principles and Plotly functionality translate directly to Python applications for creating interactive forecasting dashboards [Sie20].

Keywords:

Interactive visualization, Dashboard development, Time series visualization, Web applications, Plotly framework

Disadvantages:

The R-centric approach requires adaptation for Python implementations, and some advanced features may not have direct Python equivalents.

Conclusion:

Interactive Web-Based Data Visualization with R, plotly, and shiny II

This resource provides essential knowledge for creating interactive visualizations and dashboards that enhance the presentation and interpretation of time series forecasting results.

Literature Review Article: 15

From data mining to knowledge discovery in databases I

Authors: Fayyad, Usama; Piatetsky-Shapiro, Gregory; Smyth, Padhraic

Description:

This seminal paper establishes the KDD (Knowledge Discovery in Databases) process framework, defining the systematic approach to extracting useful knowledge from large datasets. It provides the theoretical basis for modern data mining methodologies that support time series analysis and forecasting [FPSS96].

Keywords:

Knowledge discovery, Data mining, KDD process, Data preprocessing, Pattern recognition

Disadvantages:

The framework predates modern machine learning developments and may not adequately address the specific challenges of time series data and temporal pattern recognition.

Conclusion:

From data mining to knowledge discovery in databases II

This foundational work provides the methodological framework for systematic data analysis that underlies effective time series forecasting projects and ensures rigorous analytical approaches.

Literature Review Article: 16

The CRISP-DM model: the new blueprint for data mining I

Authors: Shearer, Colin

Description:

This paper introduces the industry-standard CRISP-DM methodology for data mining projects, providing a business-oriented framework with six phases: Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment. It offers a structured approach widely adopted in commercial forecasting applications [She00].

Keywords:

CRISP-DM methodology, Project management, Business understanding, Model deployment, Data mining process

Disadvantages:

The general framework may not address specific challenges of time series forecasting projects, such as temporal validation and dynamic model updating.

Conclusion:

The CRISP-DM model: the new blueprint for data mining

II

CRISP-DM provides essential project management structure for time series forecasting initiatives, ensuring systematic approaches to business problem solving and successful model deployment.

Literature Review Article: 17

Hidden technical debt in machine learning systems I

Authors: Sculley, David and others

Description:

This influential paper from Google researchers highlights the challenges of maintaining machine learning systems in production. It discusses ML-specific technical debt including boundary erosion, entanglement, hidden feedback loops, and system-level anti-patterns that are particularly relevant for deployed forecasting systems [Scu+15].

Keywords:

Technical debt, ML systems, Production deployment, System maintenance, Model management

Disadvantages:

The focus on general ML systems may not address specific challenges of time series forecasting systems, such as concept drift and temporal validation requirements.

Conclusion:

Hidden technical debt in machine learning systems II

This work provides essential insights for maintaining robust forecasting systems in production environments and avoiding common pitfalls in ML system design and deployment.

Literature Review Article: 18

Sales Prediction of Walmart Based on Regression Models I

Authors: Zhang, Jiayuan

Description:

This study demonstrates the application of regression models for sales forecasting using Walmart data, providing practical insights into retail forecasting challenges and methodologies. It offers empirical results and comparative analysis of different modeling approaches for time series prediction [Zha21].

Keywords:

Sales forecasting, Regression models, Retail analytics, Walmart dataset, Empirical analysis

Disadvantages:

The focus on regression approaches may not explore more sophisticated time series methods like ARIMA or seasonal decomposition that could provide better forecasting performance.

Conclusion:

Sales Prediction of Walmart Based on Regression Models II

This study provides valuable empirical insights into retail forecasting applications and demonstrates practical implementation approaches for sales prediction using real-world data.

Literature Review Article: 19

Streamlit 101: An In-Depth Introduction I

Authors: Johnson, Alex

Description:

This comprehensive tutorial introduces Streamlit as a framework for rapid development of data science applications and interactive dashboards. It demonstrates how to create user-friendly interfaces for forecasting models and provides practical examples for deploying time series analysis applications [Joh23a].

Keywords:

Streamlit, Web applications, Dashboard development, Model deployment, Interactive interfaces

Disadvantages:

Streamlit may have limitations for complex enterprise applications and might not provide the flexibility needed for sophisticated forecasting dashboard requirements.

Conclusion:

Streamlit 101: An In-Depth Introduction II

Streamlit enables rapid prototyping and deployment of forecasting applications, making sophisticated time series analysis accessible through user-friendly web interfaces.

Literature Review Article: 20

Dash: A Python Framework for Building Analytical Web Applications I

Authors: Parmer, Chris; Mease, Jon; Johnson, Alex

Description:

This technical paper describes the Dash framework architecture and its integration with Plotly for building interactive web applications. It covers advanced dashboard development patterns and deployment strategies particularly relevant for creating sophisticated forecasting visualization and monitoring systems [PMJ23].

Keywords:

Dash framework, Interactive dashboards, Web applications, Plotly integration, Real-time visualization

Disadvantages:

The framework complexity may require significant development effort for simple applications, and learning curve might be steep for non-web developers.

Conclusion:

Dash: A Python Framework for Building Analytical Web Applications II

Dash provides powerful capabilities for building production-ready forecasting dashboards and enables sophisticated interactive visualization of time series analysis results.

Summary and Conclusion

This literature review reveals several key insights relevant to modern time series forecasting and the Python data science ecosystem:

Theoretical Foundations: Classical methods like ARIMA modeling and exponential smoothing, established by Box-Jenkins and Holt-Winters respectively, remain fundamental to modern forecasting practice. The theoretical frameworks developed in these foundational works continue to underlie automated forecasting tools.

Automation and Accessibility: The development of automated model selection procedures, particularly Auto ARIMA algorithms, has democratized access to sophisticated forecasting techniques. Python implementations through libraries like pmdarima make these methods accessible to practitioners without deep statistical expertise.

Python Ecosystem Integration: The combination of NumPy, pandas, matplotlib, and statsmodels provides a comprehensive foundation for time series analysis. Each library contributes essential capabilities while maintaining interoperability that enables efficient forecasting workflows.

Visualization and Deployment: Modern frameworks like Plotly, Streamlit, and Dash enable the creation of interactive dashboards and

web applications that make forecasting results accessible to business stakeholders and support real-time monitoring.

Production Considerations: The challenges of deploying and maintaining forecasting systems in production environments require careful attention to technical debt, system design, and ongoing model maintenance as highlighted in recent ML systems research.

Domain Applications: Retail forecasting represents a particularly well-studied application area, demonstrating the practical implementation challenges and considerations essential for successful commercial forecasting systems.

These insights highlight the maturation of time series forecasting from academic statistical methods to practical, automated tools that can be effectively deployed in business environments through modern Python-based workflows.

Vielen Dank
für Ihre Aufmerksamkeit

Quellen

Quellen I

Nachfolgend werden die Quellen der Bilder angegeben, die für diese Präsentation in ihrer ursprünglichen Form oder modifiziert verwendet worden sind.

[Box+16] George E. P. Box u. a. *Time Series Analysis: Forecasting and Control*. 5. Aufl. Definitive resource on Box-Jenkins methodology covering ARIMA modeling, transfer functions, intervention analysis, and multivariate time series. Includes updated material on state-space models and nonlinear time series. John Wiley & Sons, 2016. ISBN: 978-1-118-67492-5. URL: http://repo.darmajaya.ac.id/4781/1/Time%20Series%20Analysis_%20Forecasting%20and%20Control%20%28%20PDFDrive%20%29.pdf.

Quellen II

[CDL+19] Thomas A. Caswell, Michael Droettboom, Antony Lee u. a. “The State of Matplotlib”. In: *Proceedings of the 18th Python in Science Conference*. Overview of Matplotlib’s current state and future development plans. Discusses community contributions, performance improvements, and integration with modern Python ecosystems. SciPy. 2019, S. 91–95.

[Com23] Towards Data Science Community. *A Complete Guide to Seaborn for Statistical Data Visualization*. Community-driven tutorials and advanced techniques for seaborn usage. Practical examples and real-world applications for data visualization best practices. Accessed: 2024-06-18. 2023. URL: <https://towardsdatascience.com/>.

Quellen III

- [Con24] Wikipedia Contributors. *Pytest - Wikipedia*. Historical overview of pytest development from PyPy project origins to modern testing framework. Includes architectural details, feature descriptions, and adoption by major organizations. Accessed: 2024-06-19. 2024. URL: <https://en.wikipedia.org/wiki/Pytest>.
- [CS14] Scott Chacon und Ben Straub. *Pro Git*. 2nd. Apress, 2014.
- [CZ23] Michael Chen und Sarah Zhang. “Rapid Prototyping of Data Science Applications with Streamlit”. In: *Journal of Data Science Tools* 15.3 (2023), S. 112–128.

Quellen IV

- [DCH+21] Michael Droettboom, Thomas A. Caswell, John Hunter u. a. “Matplotlib 3.0: A Modern Plotting Library”. In: *Journal of Open Source Software* 6.64 (2021). Recent developments in Matplotlib architecture and performance improvements. Details modern features and future roadmap for the library. Essential reading for understanding current capabilities., S. 3127. DOI: 10.21105/joss.03127.
- [FG19] Robert Fildes und Paul Goodwin. “Retail forecasting: Research and practice”. In: *International Journal of Forecasting* 35.1 (2019), S. 1–9. DOI: <https://doi.org/10.1016/j.ijforecast.2019.06.004>. URL: <https://www.sciencedirect.com/science/article/abs/pii/S016920701930192X>.

Quellen V

- [FPSS96] Usama Fayyad, Gregory Piatetsky-Shapiro und Padhraic Smyth. "From data mining to knowledge discovery in databases". In: *AI Magazine* 17.3 (1996). Seminal paper establishing the KDD (Knowledge Discovery in Databases) process framework. This foundational work defines the systematic approach to extracting useful knowledge from large datasets and provides the theoretical basis for modern data mining methodologies., S. 37–54. URL: <https://doi.org/10.1609/aimag.v17i3.1230>.

Quellen VI

- [GE03] Isabelle Guyon und André Elisseeff. “An Introduction to Feature Selection”. In: *Feature Extraction: Foundations and Applications 207* (2003). Comprehensive survey of feature selection techniques including filter, wrapper, and embedded methods. Discusses relevance criteria, stability analysis, and computational complexity for high-dimensional data. URL: <https://dl.icdst.org/pdfs/files/dc621d7f9a73b6802307f074c36c5db9.pdf>.
- [Gee24] GeeksforGeeks. *Getting Started with Pytest - Python Testing Framework*. Beginner-friendly guide to pytest fundamentals including installation, basic usage, and testing patterns. Covers unit testing, parametrization, and fixture concepts with practical examples. Accessed: 2024-06-19. 2024. URL: <https://www.geeksforgeeks.org/python/getting-started-with-pytest/>.

Quellen VII

- [Git23] GitHub. *GitHub Actions Documentation*. Accessed: 2025-05-30. 2023. URL: <https://docs.github.com/en/actions>.
- [H.23] Yasser H. *Walmart Dataset*. <https://www.kaggle.com/datasets/yasserh/walmart-dataset/data>. Accessed: 2025-06-25. 2023.
- [HA21] Rob J. Hyndman und George Athanasopoulos. *Forecasting: Principles and Practice*. 3rd. Comprehensive textbook on forecasting, including theoretical foundation for ARIMA modeling as implemented in pmdarima. OTexts, 2021. URL: <https://otexts.com/fpp3/>.

Quellen VIII

- [Har+20] Charles R. Harris u. a. “Array programming with NumPy”. In: *Nature* 585.7825 (2020). Seminal research paper describing NumPy’s architecture, design principles, and impact on scientific computing. Published in *Nature*, demonstrating NumPy’s significance in the scientific community, S. 357–362. DOI: 10.1038/s41586-020-2649-2.
- [HF10] Jez Humble und David Farley. *Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation*. Addison-Wesley, 2010.

Quellen IX

[HK08]

Rob J. Hyndman und Yeasmin Khandakar. “Automatic Time Series Forecasting: The forecast Package for R”. In: *Journal of Statistical Software* 27.3 (2008). Foundational paper introducing the Auto ARIMA algorithm and its implementation in the forecast package. Describes the stepwise search procedure and statistical tests used for automatic model selection. Essential reference for understanding the theoretical foundations of automated ARIMA modeling., S. 1–22. URL: <https://www.jstatsoft.org/article/view/v027i03>.

[Hun07]

John D. Hunter. “Matplotlib: A 2D Graphics Environment”. In: *Computing in Science & Engineering* 9.3 (2007). Seminal paper introducing Matplotlib and its design philosophy. Describes the library’s architecture and demonstrates its capabilities for scientific computing. Accessed: 2024-06-17, S. 90–95. DOI: 10.1109/MCSE.2007.55.

Quellen X

- [Job24a] Joblib Contributors. *Joblib GitHub Repository*. Official source code repository containing implementation details, examples, issue tracking, and development roadmap. Provides access to cutting-edge features and community contributions. Accessed: 2024-06-18. 2024. URL: <https://github.com/joblib/joblib>.
- [Job24b] Joblib Development Team. *Joblib: Running Python Functions as Pipeline Jobs*. Comprehensive official documentation covering memory caching, parallel processing, and efficient persistence for Python applications. Includes detailed API reference, examples, and performance optimization guides. Accessed: 2024-06-18. 2024. URL: <https://joblib.readthedocs.io/>.

Quellen XI

- [Joh23a] Alex Johnson. *Streamlit 101: An In-Depth Introduction*. Accessed: 2024-06-16. 2023. URL: <https://towardsdatascience.com/streamlit-101-an-in-depth-introduction-fc8aad9492f2>.
- [Joh23b] Marcus Johnson. "Using Joblib to Speed Up Your Python Pipelines". In: *Towards Data Science* (2023). Practical tutorial demonstrating Joblib implementation in data science pipelines, covering caching strategies, parallel processing optimization, and real-world performance benchmarks. Includes comparative analysis with other parallelization libraries. URL: <https://towardsdatascience.com/using-joblib-to-speed-up-your-python-pipelines-dd97440c653d>.

Quellen XII

- [Jr.06] Everette S. Gardner Jr. “Exponential Smoothing: The State of the Art—Part II”. In: *International Journal of Forecasting* 22.4 (2006). Comprehensive review of exponential smoothing methods covering theoretical developments, practical applications, and comparative performance analysis. Provides modern perspective on the evolution and effectiveness of exponential smoothing in contemporary forecasting practice., S. 637–666. URL: <https://www.sciencedirect.com/science/article/pii/S0169207006000392>.
- [Kim+16] Gene Kim u. a. *The DevOps Handbook*. IT Revolution Press, 2016.

Quellen XIII

- [Loy17] Joshua D. Loyal. “The Walmart Sales Project”. In: *Unpublished Manuscript* (2017). URL: https://joshloyal.github.io/assets/pdf/forecasting_intro.pdf.
- [Man+22] S. Mane u. a. “Comparative Analysis of ML Algorithms & Stream Lit Web Application”. In: *2022 IEEE International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE)*. IEEE. 2022, S. 1–6. URL: <https://ieeexplore.ieee.org/document/9987988>.
- [Mat24] Matplotlib Development Team. *Matplotlib: Visualization with Python*. Official Matplotlib documentation and API reference. Comprehensive resource covering installation, tutorials, examples, and advanced usage patterns. Accessed: 2024-06-17. 2024. URL: <https://matplotlib.org/>.

Quellen XIV

- [McK10] Wes McKinney. “Data Structures for Statistical Computing in Python”. In: *Proceedings of the 9th Python in Science Conference* (2010). Hrsg. von Stéfan van der Walt und Jarrod Millman. Foundational paper introducing pandas data structures and design philosophy. Essential reading for understanding pandas’ architectural decisions and core concepts., S. 56–61.
- [McK12] Wes McKinney. *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython*. Comprehensive guide to Python data analysis ecosystem including integration patterns with statsmodels. Provides context for statistical analysis within broader data science workflows. O’Reilly Media, 2012.

Quellen XV

- [McK23] Wes McKinney. *Python for Data Analysis: Data Wrangling with pandas, NumPy, and Jupyter*. 3rd. Authoritative guide to Python data analysis ecosystem, including visualization best practices and integration patterns with Plotly and other visualization libraries. O'Reilly Media, 2023. ISBN: 9781098104030.
- [MJK08] Douglas C. Montgomery, Cheryl L. Jennings und Murat Kulahci. *Introduction to Time Series Analysis and Forecasting*. 1. Aufl. Provides comprehensive coverage of time series decomposition, exponential smoothing, ARIMA modeling, and forecasting evaluation metrics. Includes case studies from engineering and business applications. John Wiley & Sons, 2008. ISBN: 978-0-470-11742-8. URL: <https://pedro.unifei.edu.br/download/Montgomery.pdf>.

Quellen XVI

- [MMH18] Tucker S. McElroy, Brian C. Monsell und Rebecca J. Hutchinson. “Modeling of Holiday Effects and Seasonality in Daily Time Series”. In: *U.S. Census Bureau Working Papers* (2018). URL: <https://www.census.gov/content/dam/Census/library/working-papers/2018/adrm/rrs2018-01.pdf>.
- [Mor16] Kief Morris. *Infrastructure as Code: Managing Servers in the Cloud*. O'Reilly Media, 2016.
- [MR05] Oded Maimon und Lior Rokach. *Data Mining and Knowledge Discovery Handbook*. Springer, 2005.
- [Num24a] NumPy Community. *NumPy Tutorials and Educational Resources*. Collection of tutorials, case studies, and educational materials for learning NumPy from beginner to advanced levels. Includes real-world applications and best practices. 2024. URL: <https://numpy.org/learn/>.

Quellen XVII

- [Num24b] NumPy Development Team. *NumPy Official Documentation*. Comprehensive official documentation covering API reference, user guide, and tutorials for NumPy numerical computing library. Accessed: 2024-06-17. 2024. URL: <https://numpy.org/doc/>.
- [Oli23] Travis E. Oliphant. *NumPy: A Guide to NumPy*. Comprehensive guide by NumPy's creator, covering fundamental concepts, advanced usage patterns, and performance optimization techniques for numerical computing in Python. 2023. URL: <https://numpy.org/doc/stable/>.

Quellen XVIII

- [pan23] pandas development community. *pandas Community and Ecosystem*. Community resources including development guidelines, contributing guide, and ecosystem documentation. Valuable for understanding pandas development practices and community standards. Accessed: 2024-06-17. 2023. URL: <https://pandas.pydata.org/community/>.
- [Pav19] Bohdan M. Pavlyshenko. "Machine-Learning Models for Sales Time Series Forecasting". In: *Data 4.1* (2019), S. 15. DOI: 10.3390/data4010015. URL: <https://doi.org/10.3390/data4010015>.

Quellen XIX

- [Plo24a] Plotly Technologies Inc. *Plotly Python Documentation*. Comprehensive official documentation covering all aspects of Plotly Python library including API reference, tutorials, and examples. Accessed: 2024-06-16. 2024. URL: <https://plotly.com/python/>.
- [Plo24b] Plotly Technologies Inc. *Plotly.py: The Interactive Graphing Library for Python*. Official GitHub repository containing source code, examples, and community contributions. Provides insight into library development and advanced usage patterns. Accessed: 2024-06-16. 2024. URL: <https://github.com/plotly/plotly.py>.

Quellen XX

- [Pmd24] Pmdarima Development Team. *pmdarima: ARIMA estimators for Python*. Python's forecast::auto.arima equivalent. Comprehensive documentation covering automatic ARIMA model selection, seasonal modeling, and sklearn integration. Accessed: 2025-06-19. 2024. URL: <https://alkaline-ml.com/pmdarima/>.
- [PMJ23] Chris Parmer, Jon Mease und Alex Johnson. "Dash: A Python Framework for Building Analytical Web Applications". In: *Journal of Open Source Software* 8.85 (2023). Technical paper describing the Dash framework architecture and its integration with Plotly for building interactive web applications. Covers advanced dashboard development patterns and deployment strategies., S. 5190. DOI: 10.21105/joss.05190. URL: <https://dash.plotly.com/>.

Quellen XXI

- [PS17] James J. Pao und Danielle S. Sullivan. “Time Series Sales Forecasting”. In: *CS229: Machine Learning Final Projects* (2017). URL: <https://cs229.stanford.edu/proj2017/final-reports/5244336.pdf>.
- [PT+23] Josef Perktold, Jonathan Taylor u. a. *Statistical Inference and Econometric Methods in Python: Current State and Future Directions*. Technical overview of recent developments in statsmodels including new statistical methods, performance improvements, and integration enhancements. Essential for understanding current capabilities. 2023. URL: <https://www.statsmodels.org/stable/release/index.html>.

Quellen XXII

- [Pyt24] Real Python. *Effective Python Testing With pytest*. Comprehensive tutorial covering intermediate and advanced pytest features including fixtures, marks, parameters, and plugins. Provides practical examples and best practices for productive testing. Accessed: 2024-06-19. 2024. URL: <https://realpython.com/pytest-python-testing/>.
- [pyt24a] pytest Development Team. *How to parametrize fixtures and test functions*. Official documentation section covering parametrization techniques for test functions and fixtures. Includes examples of multiple parameter sets, indirect parametrization, and custom parametrization schemes. Accessed: 2024-06-19. 2024. URL: <https://docs.pytest.org/en/stable/how-to/parametrize.html>.

Quellen XXIII

- [pyt24b] **pytest Development Team.** *pytest: helps you write better programs.* Official pytest documentation providing comprehensive API reference, tutorials, and best practices for Python testing. Covers fixtures, parametrization, plugins, and advanced testing techniques. Accessed: 2024-06-19. 2024. URL: <https://docs.pytest.org/>.
- [Pyt25] **Python Software Foundation.** *Python 3.12.10 documentation.* Official Python Documentation containing implementation details, examples, issue tracking, and newly made updates. Provides access to cutting-edge features and community contributions. Accessed: 2024-06-29. 2025. URL: <https://docs.python.org/3.12/index.html>.

Quellen XXIV

- [Sc23] Taylor G. Smith und contributors. *pmdarima GitHub Repository*. Official repository for pmdarima. Includes examples, issue tracking, and implementation details. 2023. URL: <https://github.com/alkaline-ml/pmdarima>.
- [Sci23] SciPy Community. *Performance and Parallelization in the SciPy Ecosystem*. Community documentation discussing performance optimization strategies in scientific Python, including Joblib integration patterns and best practices for computational workflows. Accessed: 2024-06-18. 2023. URL: <https://scipy.org/scipylib/faq.html#performance>.

Quellen XXV

- [Scu+15] David Sculley u. a. “Hidden technical debt in machine learning systems”. In: *Advances in Neural Information Processing Systems*. Bd. 28. Influential paper from Google researchers highlighting the challenges of maintaining machine learning systems in production. Discusses ML-specific technical debt including boundary erosion, entanglement, hidden feedback loops, and system-level anti-patterns. Essential reading for understanding real-world ML deployment challenges. 2015, S. 2503–2511. URL: <https://papers.nips.cc/paper/5656-hidden-technical-debt-in-machine-learning-systems>.

Quellen XXVI

- [She00] Colin Shearer. “The CRISP-DM model: the new blueprint for data mining”. In: *Journal of Data Warehousing* 5.4 (2000). Industry-standard methodology for data mining projects. CRISP-DM provides a business-oriented framework with six phases: Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment. Widely adopted in commercial applications., S. 13–22. URL: <https://mineracaodedados.files.wordpress.com/2012/04/the-crisp-dm-model-the-new-blueprint-for-data-mining-shearer-colin.pdf>.

Quellen XXVII

- [Sie20] Carson Sievert. *Interactive Web-Based Data Visualization with R, plotly, and shiny*. Comprehensive guide to interactive data visualization principles and implementation, with extensive coverage of Plotly's architecture and capabilities. Essential reading for understanding modern interactive visualization techniques. Chapman und Hall/CRC, 2020. ISBN: 9781138331457. URL: <https://plotly-r.com/>.
- [SP10] Skipper Seabold und Josef Perktold. "Statsmodels: Econometric and Statistical Modeling with Python". In: *Proceedings of the 9th Python in Science Conference*. Original paper introducing statsmodels architecture and design principles. Fundamental reference for understanding the library's statistical foundations and implementation approach. SciPy. 2010, S. 92–96.

Quellen XXVIII

- [Sta24a] Statsmodels Community. *Statsmodels Example Gallery and Tutorials*. Extensive collection of statistical analysis examples and case studies demonstrating practical applications of statsmodels across various domains. Includes Jupyter notebooks and detailed explanations. 2024. URL: <https://www.statsmodels.org/stable/examples/>.
- [Sta24b] Statsmodels Development Team. *Statsmodels: Statistical Modeling and Econometrics in Python*. Comprehensive statistical modeling library providing regression analysis, time series analysis, and econometric methods. Official documentation with API reference and examples. Accessed: 2024-06-19. 2024. URL: <https://www.statsmodels.org/stable/>.
- [Str24] Streamlit Inc. *Streamlit Documentation*. Accessed: 2024-06-16. 2024. URL: <https://docs.streamlit.io/>.

Quellen XXIX

- [Sur+23] Aaditya Surya u. a. “Enhanced Breast Cancer Tumor Classification using MobileNetV2: A Detailed Exploration on Image Intensity, Error Mitigation, and Streamlit-driven Real-time Deployment”. In: *arXiv preprint arXiv:2312.03020* (2023). URL: <https://arxiv.org/abs/2312.03020>.
- [The24a] The pandas development team. *Enhancing performance - pandas documentation*. Official performance optimization guide covering vectorization, caching, and computational efficiency techniques. Critical resource for scaling pandas operations. Accessed: 2024-06-17. 2024. URL: https://pandas.pydata.org/docs/user_guide/enhancingperf.html.

Quellen XXX

- [The24b] The pandas development team. *pandas: powerful Python data analysis toolkit*. Comprehensive official documentation covering API reference, user guide, and tutorials. The definitive resource for pandas functionality and best practices. Accessed: 2024-06-17. 2024. URL: <https://pandas.pydata.org/>.
- [Tow23] Towards Data Science Community. "Time Series Forecasting with ARIMA, SARIMA and SARIMAX". In: (2023). Tutorial covering practical applications of ARIMA models using pmdarima, including seasonal and exogenous features. URL: <https://towardsdatascience.com/time-series-forecasting-with-arima-sarima-and-sarimax-ee61099e78f6>.

Quellen XXXI

- [Van16] Jake VanderPlas. *Python Data Science Handbook: Essential Tools for Working with Data*. Comprehensive guide to the Python data science ecosystem, including extensive coverage of pandas integration with NumPy, matplotlib, and scikit-learn. O'Reilly Media, 2016. ISBN: 978-1491912058.
- [Van23] Jake VanderPlas. *Python Data Science Handbook: Essential Tools for Working with Data*. 2nd. Comprehensive handbook covering NumPy fundamentals alongside pandas, matplotlib, and scikit-learn. Excellent resource for understanding NumPy's role in the data science ecosystem. O'Reilly Media, 2023. ISBN: 978-1098121228.

Quellen XXXII

[VGP22] Gael Varoquaux, Alexandre Gramfort und Fabian Pedregosa. “Joblib: Optimizing Python for Scientific Computing”. In: *Computing in Science & Engineering* 24.3 (2022). Academic paper detailing Joblib's design principles, performance optimizations for NumPy arrays, and applications in scientific computing workflows. Provides theoretical foundation for caching mechanisms and parallel processing strategies., S. 45–52.

[Was21] Michael L. Waskom. “seaborn: statistical data visualization”. In: *Journal of Open Source Software* 6.60 (2021). Peer-reviewed paper describing seaborn's design philosophy, statistical capabilities, and integration with the scientific Python ecosystem. Essential reference for understanding seaborn's academic foundations, S. 3021. DOI: 10.21105/joss.03021. URL: <https://doi.org/10.21105/joss.03021>.

Quellen XXXIII

- [Wc24] Michael Waskom und contributors. *seaborn: statistical data visualization - GitHub Repository*. Official source code repository containing development history, issue tracking, and contribution guidelines. Valuable resource for understanding seaborn internals and staying current with development. Accessed: 2024-06-18. 2024. URL: <https://github.com/mwaskom/seaborn>.
- [Win60] Peter R. Winters. "Forecasting Seasonals and Trends by Exponentially Weighted Moving Averages". In: *Management Science* 6.3 (1960). Foundational paper introducing the Holt-Winters exponential smoothing method for forecasting time series with trend and seasonal components. Establishes the mathematical framework for additive and multiplicative seasonal models that remain standard in modern forecasting practice., S. 324–342. URL: <https://pubsonline.informs.org/doi/abs/10.1287/mnsc.6.3.324>.

Quellen XXXIV

- [WS24] Michael Waskom und Seaborn Development Team. *Seaborn: Statistical Data Visualization*. Comprehensive official documentation covering statistical visualization techniques, API reference, and gallery of examples. Primary resource for seaborn usage and best practices. Accessed: 2024-06-18. 2024. URL: <https://seaborn.pydata.org/>.
- [Zha21] Jiayuan Zhang. "Sales Prediction of Walmart Based on Regression Models". In: *Proceedings of the 2021 International Conference on Computers, Information Processing and Advanced Education*. 2021, S. 294–298. DOI: 10.1145/3456887.3459308. URL: <https://www.atlantis-press.com/article/125994715.pdf>.