TU WIEN Informatics

Temporal Analysis of Session Clusters in Clickstream Data from a Price Comparison Platform

BACHELOR'S THESIS

submitted in partial fulfillment of the requirements for the degree of

Bachelor of Science

in

Informatics

by

Luca Turin
Registration Number 12220876

to the Faculty of Informatics

at the TU Wien

Advisor: Univ.Ass.in Mag.a rer.nat. Dr.in techn. Julia Neidhardt
Assistance: Dipl.-Ing. Ahmadou Wagne, B.A.

Vienna, January 31, 2026

Luca Turin	Julia Neidhardt

Declaration of Authorship

Luca Turin

I hereby declare that I have written this Bachelor's Thesis independently, that I have completely specified the utilized sources and resources and that I have definitely marked all parts of the work - including tables, maps and figures - which belong to other works or to the internet, literally or extracted, by referencing the source as borrowed.
I further declare that I have used generative AI tools only as an aid, and that my own intellectual and creative efforts predominate in this work. In the appendix "Overview of Generative AI Tools Used" I have listed all generative AI tools that were used in the creation of this work, and indicated where in the work they were used. If whole passages of text were used without substantial changes, I have indicated the input (prompts) I formulated and the IT application used with its product name and version number/date.

Vienna, January 31, 2026

Luca Turin

Acknowledgements

At this point, I would like to express my sincere gratitude to Julia Neidhardt and Ahmadou Wagne for their invaluable guidance and expertise in supervising this thesis.

I am deeply thankful to my mother and father for their endless support throughout my studies.

Finally, I would like to sincerely thank all my friends, both old and new, for their constant encouragement and for always reminding me to never give up and *never lie down.*

Abstract

Understanding user behaviour is essential for designing digital services, improving user experience, and optimising commercial outcomes. This thesis investigates how behaviour varies over time and across contexts on a major price comparison platform. Building on prior work that offered a static segmentation of sessions, it examines how these patterns evolve over time. The study addresses three questions about the extent of behavioural differences: month to month, weekday versus weekend, and shifts during the Black Friday period.

We analyse one year of clickstream data from July 2024 to June 2025. We segment the data by time, apply unsupervised clustering within each segment, and align labels across segments to enable comparison.

The results show a stable set of behavioural segments together with clear temporal change. Month to month differences are evident, with a shift toward deal seeking during the end of year shopping period and around Black Friday. Weekends show more exploratory behaviour than weekdays. Black Friday polarises activity, with more short check in visits and more concentrated deal focused research. These patterns indicate that user behaviour changes meaningfully with time and context.

Contents

CHAPTER 1

Introduction

1.1 Motivation

As online platforms continue to grow and diversify, understanding user behaviour has become essential for designing effective digital services, improving user experience, and optimizing commercial outcomes. A particularly valuable source of insight in this context is clickstream data, which captures the sequence of interactions a user performs while navigating a website. These interactions, such as product views, searches, or filter changes are typically grouped into sessions, representing a single visit to the platform. Analyzing such sessions can reveal how visitors explore the site, what they are interested in, and how different usage patterns emerge [34].

In recent years, a large body of research has focused on the analysis of clickstream data from traditional e-commerce platforms, where users not only browse but also make purchases directly [15, 57, 44, 34]. In these studies, the main goal is to understand purchasing behaviour, predict conversion, or identify customer segments with high revenue potential.

Some studies have also explored the question of how user behaviour changes over time [35, 30]. These works investigate, for example, whether repeat visitors tend to shop more efficiently, or how engagement patterns evolve as users become more familiar with a platform. However, the majority of contributions still focus on classic e-commerce environments.

Price-comparison platforms represent a different but equally important category of online services. Unlike e-commerce stores, these platforms do not facilitate purchases directly. Instead, they aggregate product information and prices from multiple online retailers, enabling users to make informed decisions and redirecting them to third-party sites for

the actual transaction. Geizhals[1] is one such platform, widely used in German-speaking countries and operating also in the UK and Poland.

Price-comparison platforms have received little attention in clickstream research. In particular, there is a lack of studies that examine how user behaviour on such platforms evolves over time. This thesis aims to address this gap by analyzing clickstream sessions on Geizhals with a specific focus on temporal patterns.

By applying clustering to sessions across different time windows, comparing the resulting clusters, and visualising these dynamics using interactive dashboards (e.g., Grafana[2]), this work aims to contribute to a better understanding of temporal dynamics in session behaviour, particularly in the context of non-transactional platforms.

Furthermore, while many clickstream analyses in commercial settings focus on short-term KPIs such as conversion or bounce rates [36, 4], this thesis takes a more exploratory perspective by examining underlying behavioural logics. The aim is to uncover patterns that may not directly reflect immediate business outcomes, but which could have longer-term relevance for platform strategy, user segmentation, or interface design [50].

This thesis extends the work of Wagne and Neidhardt, who identified distinct session types on price comparison platforms using anonymized clickstream data from Geizhals [53]. While their study provides a static segmentation of user sessions, this work focuses on uncovering how these behavioural patterns evolve over time. To this end, temporal clustering in this thesis refers to the repeated application of static unsupervised clustering to temporally segmented data, rather than to sequence-based or time-aware clustering models. This approach allows for detecting changes in session types over time by comparing the resulting cluster solutions for different time windows.

Building upon these considerations, this thesis makes the following contributions to the analysis of user behaviour on price-comparison platforms:

- **Temporal and contextual analysis of user behaviour:** This thesis investigates how session-level behavioural patterns evolve over time and in specific contexts. It extends previous static clustering approaches by applying unsupervised clustering across multiple time windows to detect temporal dynamics in user behaviour.
- **Exploration beyond short-term KPIs:** While many commercial studies focus on immediate performance metrics, this work adopts a more exploratory approach to uncover deeper behavioural patterns.
- **Implementation of an interactive monitoring dashboard:** Beyond the analysis itself, this thesis provides a functional Grafana-based dashboard. This tool enables the observation and interpretation of session distributions and cluster evolution, making complex behavioural dynamics accessible to non-technical stakeholders and supporting data-driven decision-making.

[1] https://geizhals.at/
[2] https://grafana.com/

1.2 Research Questions and Objectives

This thesis investigates whether behavioural patterns in user sessions on a price-comparison platform change over time and in specific contextual situations, such as promotional periods. The focus lies on comparing session types using unsupervised clustering across multiple time windows.

The central research question is:

> *What temporal and contextual differences can be observed in session-level behavioural patterns derived through unsupervised clustering on a price-comparison platform?*

To answer this question, the following sub-questions are examined:

- **RQ1:** To what extent do session clusters differ when comparing different months throughout the year?
- **RQ2:** To what extent do session clusters differ when comparing weekdays and weekends?
- **RQ3:** To what extent do session clusters differ when comparing promotional events such as Black Friday with regular periods?

The following objectives guide the analysis:

- Apply unsupervised clustering to session data segmented by time (monthly, weekly, event-based).
- Compare the resulting clusters across time windows to identify changes in user behaviour.
- Visualize session distributions and cluster dynamics using Grafana to support interpretation.

1.3 Structure of the Thesis

This thesis is structured into six chapters, beginning with the theoretical background in Chapter 2, that discusses price comparison platforms, clickstream data, and provides an overview of relevant clustering techniques and visualisation methods. Chapter 3 introduces the data source used in this study. Building on the foundation of Chapter 2 and 3, Chapter 4 describes the methodological approach. It outlines the preprocessing steps, the session feature model, the temporal segmentation of the data, and the clustering pipeline. Chapter 5 presents the results and the discussion. It interprets the discovered

session clusters, analyses their temporal dynamics, and examines the outputs of the interactive dashboard. Finally, Chapter 6 concludes the thesis by summarising the findings, discussing their practical implications, and outlining directions for future research.

CHAPTER 2

Background

This chapter provides the theoretical and methodological background relevant to the study. It begins by introducing the context of price comparison platforms and the characteristics of clickstream data. We then introduce the session types discovered on Geizhals by Wagne and Neidhardt [53], which serve as the basis for the analysis. Additionally, a range of clustering techniques is reviewed to provide context for the methodological approach adopted in this thesis. Finally, the chapter outlines visualization strategies for understanding and interpreting clustering results.

2.1 Price Comparison Platforms

Price comparison platforms are web services that allow users to compare prices and offers for a specific product or product category from different online retailers and e-commerce shops on a single website or app. The aim of these platforms is to increase price transparency and help consumers find the best offer. They aggregate data from retailers either by cooperating with them or by searching through their product catalog. The information displayed usually includes the price, the rating/reviews of users, payment methods, the availability and shipping options.

The primary use cases for these platforms are identifying the cheapest offer for a known product, exploring available options within a given category, and researching product details before purchase. Users finalize their purchase on an external retailer's website, to which they are redirected after selecting an offer [7, 26].

2.2 Clickstream Data

Clickstream data refers to the sequence of user interactions recorded as they navigate through a website or app. Each interaction, can include a variety of actions such as page

visits, product views, searches, or clicks on external links. Clickstream logs are typically timestamped and can contain additional metadata about the page visited or the type of action performed. This data is often grouped into sessions, which represent a coherent sequence of interactions by a user within a limited time frame [29].

Clickstream data is valuable because it reflects real behaviour rather than self-reported intentions [44, 42]. However, it also presents challenges: the data sets can become very large, the events often lack explicit semantic meaning, and often sessions are anonymous since it is not possible to track the user reliably [15, 48].

In research, clickstream analysis is frequently applied in order to process and interpret user behaviour data [34]. Various methods are employed to detect behavioural patterns and develop models for personalization, clustering, or prediction [18]. As outlined in the research questions, this work applies clustering techniques to analyse user sessions.

2.3 Session Types on Geizhals

Building on the study by Wagne and Neidhardt, six recurring behavioural session types have been identified for Geizhals traffic. Although the exact cluster centroids are specific to their 2023-24 data set, the underlying intentions appear to be robust and offer a helpful conceptual basis for this study.

These session types serve a dual purpose in this thesis. First, they motivate the feature model adopted in Chapter 4 by illustrating which aspects of user behaviour are most relevant on price comparison sites.

Second, they act as a benchmark for evaluating the temporal clusters derived later in Chapter 5, enabling a comparison between previously reported patterns and the dynamics observed in our 2024-25 data. In the following, we give an overview of each of the six session types:

Quick peek. These sessions are brief check-ins, typically started from a direct link or bookmark. Users visit just one product page to check price or availability and then leave immediately. As a result, sessions are very short with minimal page depth, retailer referrals, and additional pages explored.

Major purchase. Visitors in this category have already narrowed their choice to one expensive item and are primarily interested in securing the best retailer offer. They view only a handful of product pages but generate one or more outbound clicks to compare shipping costs, warranty terms, or payment options.

Constraint-based browsing. Here, users systematically refine a broad category by adjusting filters. Filter actions and product-comparison widgets dominate user actions, whereas full-text search is rarely used.

Knowledge seeking. These sessions focus on in-depth product research without an immediate purchase intent. Users spend a long time on each page but view only a few products, indicating a targeted approach instead of broad browsing.

Query & browse. This session type combines keyword-based searching with free browsing. Users enter text queries, explore various categories, and visit deal pages. Some sessions show focused intent, while others reflect more exploratory or hedonic behaviour [34].

Heavy browsing. These sessions reflect the core user base and are defined by intense activity across every browsing dimension. Users jump between a wide range of categories, apply filters, and rely on comparison tools. The activity pattern suggests exhaustive component research, for instance when assembling a PC build or evaluating alternative configurations.

Together, these six session types illustrate the behavioural diversity present on Geizhals and motivate the feature model adopted in Chapter 4.

2.4 Clustering Techniques

Clustering is a form of unsupervised learning that aims to group a set of unlabelled data instances into clusters, such that instances in the same cluster are more similar to each other than to those in different clusters. In contrast to supervised learning, clustering does not rely on predefined class labels, but seeks to discover inherent structure or patterns in the data. In the context of clickstream data, clustering can be used to find groups of similar user sessions or navigation behaviours. This reveals hidden patterns in how users browse, which is valuable for tasks like user segmentation, recommendation, or marketing personalization [10]. Prior research has shown that clustering can uncover behavioural factors in clickstream data, including visit occasion [38], shopping motivation [24], or browsing patterns [34].

To analyze clickstream data, a variety of clustering algorithms can be applied [31, 39]. This section provides an overview of clustering techniques, starting with the widely-used **K-Means Clustering** [32], followed by several alternative methods and their characteristics: **DBSCAN** (a density-based method) [14], **Hierarchical Clustering** [23], **Gaussian Mixture Models** (a probabilistic clustering approach) [6], and **Spectral Clustering** [52]. For each technique, we outline how it works and discuss its strengths and weaknesses.

2.4.1 K-Means Clustering

K-Means clustering is one of the most popular partitioning algorithms for discovering clusters in unlabelled data. Given a desired number of clusters K, the algorithm iteratively partitions the data set into K clusters by assigning each data point to the cluster with

the nearest mean (centroid). This process aims to minimize the within-cluster variance, i.e., the sum of squared distances from each point to its cluster centroid. The standard K-Means procedure operates as follows:

1. Initialize K cluster centroids (for example, by choosing K random points).
2. Assign each data point to the nearest centroid (usually measured by Euclidean distance), forming K clusters.
3. Recompute each centroid as the mean of all points assigned to that cluster.
4. Repeat the assign-and-update steps until convergence (cluster assignments no longer change or the centroid shifts become negligible).

This simple iterative algorithm is efficient in practice, with each iteration running in $O(nK)$ time for n points and typically only a moderate number of iterations needed. K-Means has been applied very widely due to its simplicity and speed [9, 2].

However, K-Means also has important limitations. First, the value of K must be chosen in advance, and selecting an appropriate number of clusters is not trivial when the true structure of the data is unknown. The algorithm will always partition the data into K groups even if the natural structure does not neatly split into that many clusters. Second, K-Means is good at finding round, circular clusters, but has difficulty with differently structured clusters. It tends to break down when clusters have complex shapes or widely varying sizes, because every point is assigned to the nearest centroid by distance [14]. Moreover, K-Means is sensitive to outliers and noise: a single outlier can significantly shift a centroid, skewing the cluster location [51]. Another drawback is that the algorithm's result can depend on the initial centroid placements (it may converge to a local minimum of the clustering objective). Despite these limitations, its simplicity and efficiency make K-Means a common and popular starting point for clustering tasks [48].

2.4.2 Density-Based Clustering (DBSCAN)

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a clustering algorithm that identifies groups of closely packed data points while marking isolated points as noise. Unlike K-Means, it does not require the number of clusters to be specified in advance. Instead, it relies on two parameters: the neighbourhood radius ε and the minimum number of points (`minPts`) needed to form a dense region.

The algorithm works by selecting an arbitrary point and exploring its neighbourhood. If there are enough nearby points (at least `minPts` within ε), it becomes the core of a new cluster. This cluster is then expanded by recursively including neighbours that also satisfy the density condition. Points that do not belong to any dense region are labelled as noise.

DBSCAN has several advantages over traditional clustering methods. It can detect clusters of arbitrary shape, is robust to noise, and automatically excludes outliers without

requiring prior knowledge of the number of clusters [45]. These properties make DBSCAN particularly useful in exploratory data analysis and in domains where the underlying data distribution is unknown or complex.

However, DBSCAN also has limitations. Its performance depends heavily on the choice of ε and `minPts`, which can be difficult to tune, especially when the data has varying density. If these parameters are not chosen well, the algorithm may either split true clusters or merge unrelated sessions. In high-dimensional spaces, the notion of density also becomes less reliable [27].

In summary, DBSCAN offers a flexible and robust alternative to K-Means. It is particularly valuable when the number of clusters is unknown, when the data contains noise, or when clusters are expected to have irregular shapes.

2.4.3 Hierarchical Clustering

Hierarchical clustering is an unsupervised learning method that builds a hierarchy of clusters by either progressively merging smaller clusters (agglomerative approach) or recursively splitting larger ones (divisive approach). The result is a tree-like structure known as a dendrogram, which represents cluster relationships at multiple levels of granularity.

The agglomerative variant is more commonly used and begins by treating each data point as an individual cluster. Clusters are then successively merged based on a chosen distance or similarity criterion, such as single linkage (minimum distance), complete linkage (maximum distance), or average linkage, until all data points are grouped into a single cluster or a stopping condition is met [19].

One of the main advantages of hierarchical clustering is that it does not require the number of clusters to be specified in advance. Instead, the dendrogram allows for exploration of the data at multiple levels of granularity and the identification of meaningful cluster groupings. This makes hierarchical clustering particularly useful in exploratory data analysis, when the underlying cluster structure is not well understood. Additionally, hierarchical clustering is compatible with a wide range of distance metrics and similarity functions, including those suitable for non-numeric or structured data [22].

However, the method also has drawbacks. For large data sets it can be computationally intensive as it requires the computation and storage of a full pairwise distance matrix. Furthermore, decisions made early in the clustering process are not revisited, which can lead to suboptimal merges if noise or outliers are present in the data.

In summary, hierarchical clustering offers a flexible and interpretable framework for unsupervised learning. It is particularly useful when the number of clusters is not known in advance and when insights at different levels of granularity are desired.

2.4.4 Gaussian Mixture Models (GMM)

Gaussian Mixture Models (GMMs) offer a probabilistic framework for clustering by modelling the data as a mixture of multiple Gaussian distributions. Each component of the mixture represents a cluster, characterized by its own mean and covariance structure. The model estimates the parameters (means, covariances, and mixture weights) of the Gaussians using the Expectation-Maximization (EM) algorithm [37], an iterative method that alternates between estimating cluster membership probabilities (Expectation step) and updating the parameters to maximize the likelihood of the data (Maximization step).

Unlike K-Means, which assigns each data point to a single cluster based on proximity to a centroid, GMMs provide soft assignments, where each point is associated with a probability of belonging to each cluster. This allows for more flexible modelling of overlapping clusters and better captures uncertainty in cluster membership. Another advantage of GMMs is their ability to model clusters with elliptical shapes and varying sizes.

However, GMMs have several limitations. Like Hierarchical Clustering, the EM algorithm can be computationally intensive for large data sets and is sensitive to initialization, which may lead to convergence at local optima [5]. Moreover, in high-dimensional spaces, GMMs often struggle to accurately model the relationships between features, leading to unreliable cluster shapes, and the assumption that clusters follow a Gaussian distribution may not hold in practice with complex data [17].

In summary, GMMs are a powerful alternative to centroid-based clustering methods when the data has complex or overlapping cluster structures. Their probabilistic nature and flexibility make them suitable for a wide range of applications where soft assignments and statistical interpretability are desired.

2.4.5 Spectral Clustering

Spectral clustering uses similarities between data points to find complex groups. Instead of operating directly in the original feature space, the method begins by constructing a similarity matrix that encodes the relationships between all data points. A graph is then derived from this matrix, and the graph Laplacian is computed. Through eigenvalue decomposition of the Laplacian, the eigenvectors corresponding to the smallest non-trivial eigenvalues are selected to embed the data into a lower-dimensional space where conventional clustering algorithms such as K-Means can be applied [12].

One of the main strengths of spectral clustering is its ability to detect clusters of arbitrary shape. This is possible because spectral clustering captures global structure through the eigenvalues of the similarity graph rather than relying on local distance metrics alone.

However, spectral clustering has several limitations. It requires the number of clusters to be specified in advance and is computationally intensive for large data sets due to the need to compute and store a complete similarity matrix and its eigenvectors. Furthermore,

it does not inherently handle noise or outliers, as every data point is included in the similarity graph and influences the final clustering [12].

In summary, spectral clustering is a powerful method when the data exhibits complex or non-linear structures that are not well captured by traditional techniques.

In conclusion, clustering techniques are central to clickstream data analysis. K-Means provides a quick and simple way to partition sessions. DBSCAN offers robustness to noise and finds arbitrary-shaped clusters without needing to guess the number of segments. Hierarchical clustering offers a flexible view of group structures across multiple levels of granularity. Gaussian Mixture Models offer a statistical way to assign data points to multiple clusters, which helps capture complex patterns. Spectral clustering finds non-linear groups by using a graph that shows how similar the sessions are. Each of these techniques has strengths and weaknesses, and the choice often depends on the specific characteristics of the clickstream data at hand and the goals of the analysis [11]. In this thesis, K-Means clustering is selected as the primary method due to its scalability, interpretability, and its successful application in prior work on Geizhals clickstream data [53]. The choice of the number of clusters is discussed in more detail in Chapter 4.

2.5 Cluster Visualization and Dashboarding

Clustering results alone are rarely useful without proper interpretation. Especially in exploratory analyses like those based on clickstream data, visualizing cluster structures and presenting them in an accessible format is important for generating actionable insights [54].

This section introduces techniques to visualise the composition and behaviour of user/session clusters, with a particular focus on temporal dynamics. In addition, it describes how these visualizations can be integrated into an interactive dashboard to support monitoring, behavioural segmentation, and decision-making. The following subsections cover methods to illustrate overall cluster characteristics, and present the results through a practical dashboard implementation.

2.5.1 Visualizing Cluster Composition

Before exploring how clusters evolve over time, it is essential to understand their overall composition and internal structure. Static visualizations help interpret the characteristics of each cluster and highlight differences in user behaviour [55].

A common approach is to use bar plots to display the distribution of sessions across clusters. This gives an overview of which behavioural groups are dominant in the data set. For example, a large cluster may represent typical browsing sessions, while smaller clusters could capture niche behaviours like quick purchases or heavy browsing. Figure 2.1 shows an example of such a distribution across six clusters.

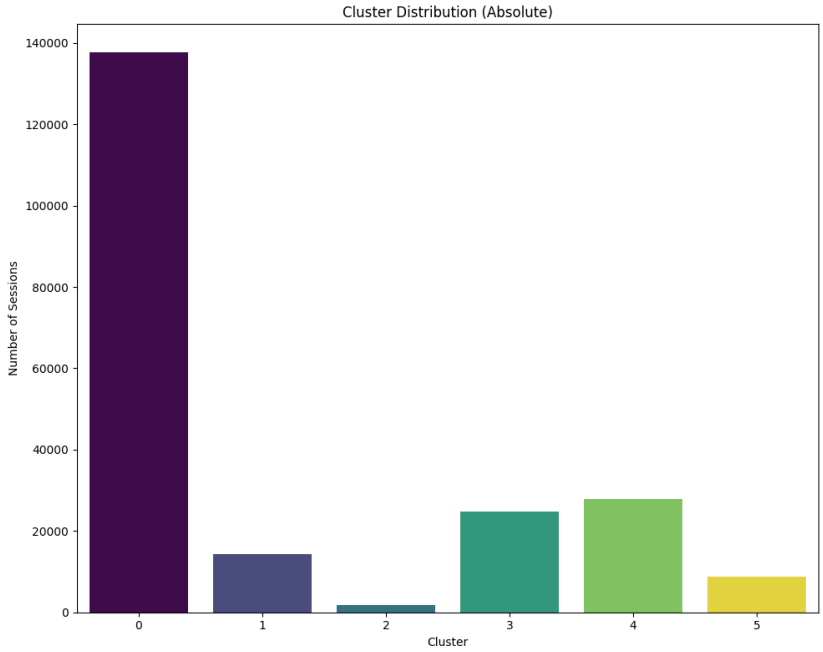


Figure 2.1: Distribution of sessions across clusters. Reproduced from [53]

To understand what differentiates the clusters, feature-based visualizations can be employed. Heat maps are particularly useful for this purpose: each column represents a cluster, and each row shows an aggregated feature such as average session duration, number of pages visited, or number of filters used. This allows for quick comparison between clusters and supports interpretation and labelling (e.g., “Quick peek”, “Major purchase”, or “Heavy browsing”). An example of such a cluster-feature heat map is shown in Figure 2.2, which visualizes standardized feature means for each cluster.

For more detailed views, distributions of selected features can be plotted per cluster using boxplots or violin plots. This reveals intra-cluster variability and makes it easier to detect overlapping behaviours or outliers [54]. An example is shown in Figure 2.3, where the standardized average product view prices are visualized per cluster.

Altogether, these techniques transform abstract cluster assignments into interpretable behavioural patterns. They also support communication with stakeholders, enabling decisions based on visualizations and plots rather than raw data alone. In this study, we focus on bar plots and heat maps, as they offer an aggregated view and are suited for determining cluster characteristics.

2.5.2 Temporal Evolution of Clusters

While static visualizations provide insights into the structure of user behaviour, clickstream data is inherently temporal. Therefore, understanding how the clusters evolve over time

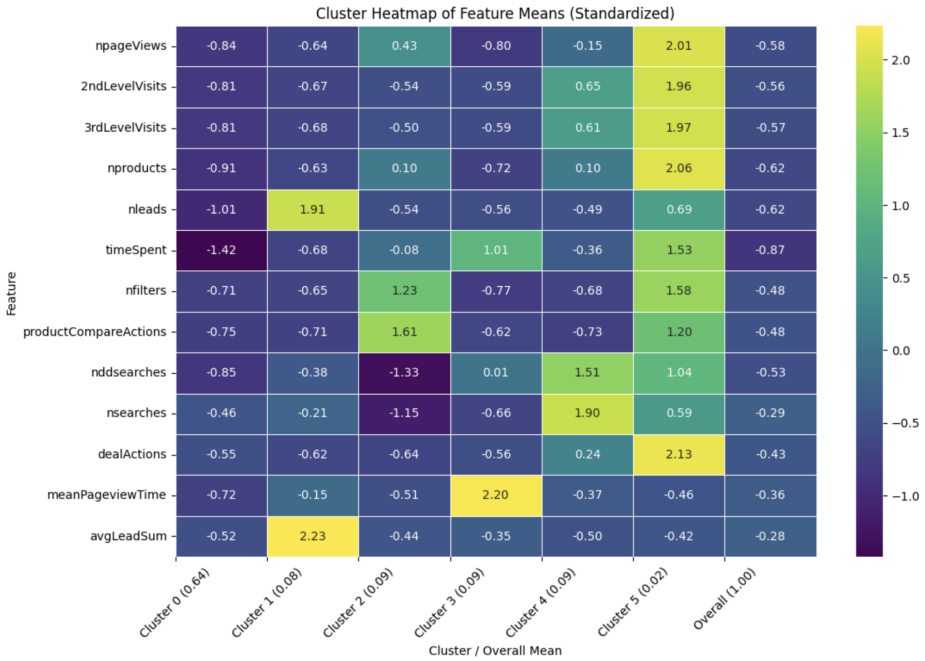


Figure 2.2: Cluster heat map of standardized feature means. Reproduced from [53]

can reveal important behavioural trends, seasonal patterns, or reactions to special events.

A first approach is a line plot that shows the total number of sessions over time. Here, the x-axis represents dates (e.g., months) and the line shows the overall traffic. This time series can highlight trends and signal specific periods. Peaks may mark the beginning of the Christmas shopping season, while dips could reflect summer holidays. Such visualisations are useful not only for behavioural segmentation, but also for forecasting or anomaly detection. For example, a sudden drop in activity in a usually dominant cluster may indicate a technical issue or a shift in user preferences [40].

A more detailed grained view can be achieved with bar plots to compare the relative size of each cluster across different time windows (e.g., by month) [56]. For instance, a comparison of September and November may show a sharp increase in certain clusters during sales periods such as Black Friday, indicating behaviour influenced by promotions. Stacking distributions side by side provides a compact overview of multiple periods, making it easier to detect seasonal drift or sudden changes quickly.

To support the temporal analysis, sessions are grouped into discrete time windows (e.g., by month), and clustering is performed separately for each window. This approach allows time-specific behavioural patterns to be identified without assuming stability in clusters over time [33]. By comparing the structure, size, and feature composition of clusters across these time windows, this approach supports the identification of behavioural shifts and the detection of time-specific usage patterns.

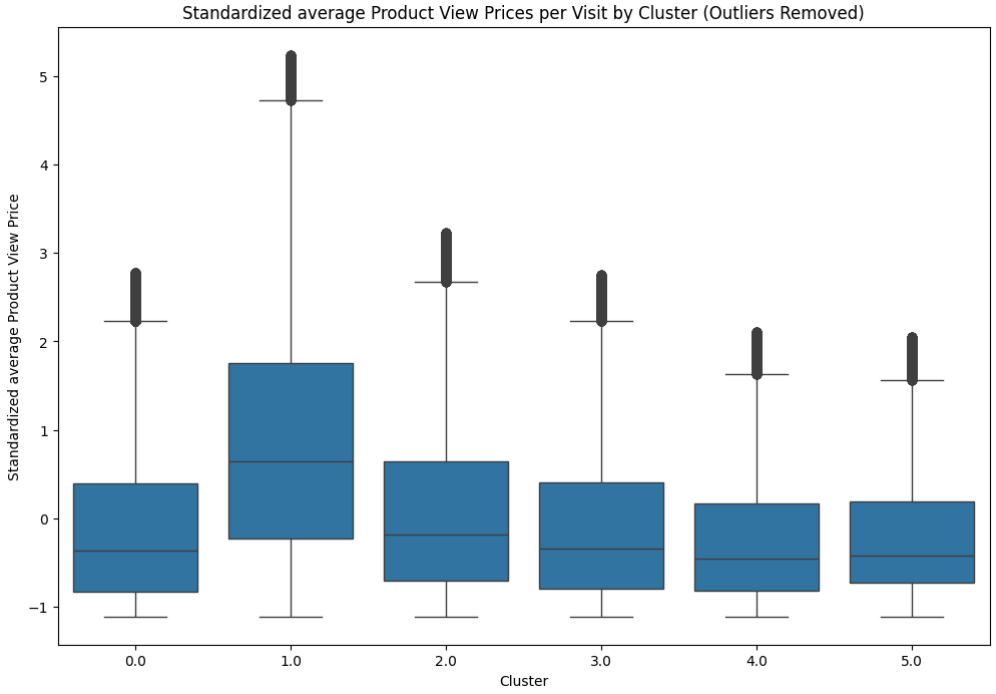


Figure 2.3: Standardised average product view price per visit by cluster. Reproduced from [53]

2.5.3 Interactive Dashboard Design

To fully leverage the insights gained through clustering, results should be made accessible in an interactive and user-friendly format. Dashboards serve this purpose by integrating visualizations and statistics into a dynamic interface that supports exploration, comparison, and decision-making. A well-designed dashboard allows analysts and stakeholders to filter data by cluster, time period, or user segment, and observe how behaviours differ across these dimensions.

Time-based visualizations, as discussed earlier, are also central to dashboarding. Embedding interactive plots for cluster activity enables users to spot trends, anomalies, or the impact of external events such as sales promotions or technical issues.

In practical terms, dashboards can be implemented using tools like Grafana, Tableau[1], or web frameworks such as Streamlit[2]. In this thesis, visualizations were exported to an interactive dashboard using Grafana.

Overall, dashboards make data analysis more accessible and interactive. They allow non-technical users to engage with clustering results intuitively [16].

[1] https://tableau.com/
[2] https://streamlit.io/

This section presented methods for visualising the results of session clustering. To understand the internal structure of each cluster, various static visualisation techniques were introduced, including bar plots, heat maps, and boxplots. These help identify dominant session types, behavioural differences between clusters, and intra-cluster variability. To capture temporal dynamics, line and bar plots were discussed to reveal trends and seasonal patterns. Finally, the section described how these visualisations can be integrated into an interactive dashboard, making the results accessible to non-technical stakeholders and useful for monitoring and decision-making.

CHAPTER 3

Data Set

The data set analyzed in this thesis is based on clickstream data collected by the web analytics platform Matomo[1], which was used to monitor user interactions. Matomo is an open-source tool that tracks user behaviour, including page views, search queries, category navigation, filter usage, product comparisons, and outbound clicks to external retailers (leads).

For this analysis, anonymized session data was exported via Matomo's API in CSV format. Since persistent user identifiers are hardly available, the exported data is organized around user sessions, where each session captures a sequence of interactions performed by a user within a continuous time frame.

Each row in the CSV files represents a single user interaction. These rows contain various dimensions such as session identifiers, timestamps, types of user actions, and product-related metadata. Table 3.1 summarizes the key columns present in the raw data set along with their descriptions.

Sessions typically consist of multiple such interactions and therefore form the basis for further feature extraction and clustering. These higher-level features are created by aggregating interaction data per session, as explained in the next chapter.

The raw Matomo export covers all recorded user activity from 1 July 2024 to 1 July 2025, spanning a full year. During this time, Geizhals recorded a total of 290,122,706 interaction events, which were aggregated into 36,649,645 user sessions prior to any filtering or preprocessing steps. The raw CSV files for this period occupy approximately 135 GB of storage.

To answer the research questions, the data is grouped by calendar month, weekday versus weekend, and the Black Friday period. These partitions are motivated and formally defined in Section 4.3.

[1] https://matomo.org/

Table 3.1: Selected columns from the raw Matomo clickstream data set

Column	Description
idVisit	Unique identifier for each session.
visitorId	Identifier for each visitor (subject to cookie policy).
visitorType	Visitor classification: new, returning, returningCustomer.
referrerType	Entry point of the session: direct, search, campaign, etc.
country	Origin country of the session.
type	Type of interaction: action, ecommerceOrder, event, search, etc.
url	Address of site on which the current action was performed.
timeSpent	Time spent on a page view (seconds).
timestamp	Unix timestamp.
pageviewPosition	Chronological order of page views within a session.
dimension1	Position within site hierarchy or type indicator (e.g., „Produktvergleich“, „Deals“).
productViewPrice	Price of viewed product.
productViewSku	ID of the viewed product.
eventCategory/Action/Name	Metadata about user-triggered events.
siteSearchKeyword	Search query used within the website.
orderId	Unique identifier of a lead.
revenue	Revenue of a lead, i.e., the product price

During the twelve-month observation period, the site averaged approximately 100,410 sessions per day. The lowest daily traffic was observed on July 10 with 55,610 sessions, while the highest occurred on November 27 with 230,252 sessions. On a monthly basis, November clearly stands out with over 4,100,000 sessions and an average of more than 137,000 sessions per day. In contrast, July and August show the lowest monthly volumes, averaging around 74,000 and 72,000 sessions per day.

When separating traffic by weekday and weekend, distinct behavioural rhythms emerge. Across the data set, 21,707,886 sessions occurred on weekdays (Monday to Thursday), corresponding to an average of 103,865 sessions per day. Weekends accounted for 9,926,519 sessions, averaging 95,447 per day.

A particularly notable spike is observed during the Black Friday promotional window, defined here as the period from 21 November to 2 December 2024. This twelve-day span recorded a total of 2,098,282 sessions, which translates to an average of 174,857 sessions per day.

CHAPTER 4

Methodology

This chapter explains how the raw Matomo clickstream logs are transformed into meaningful behavioural clusters. We first introduce the feature model, which converts event-level data into structured session-level feature vectors. Based on this representation, we then outline the clustering pipeline, beginning with preprocessing steps such as filtering and outlier removal and proceeding with dimensionality reduction and Mini-Batch K-Means clustering. The following section explains how temporal clustering is applied to address the three research questions. We then describe how consistent labelling across different time slices is ensured. Finally, we outline the metrics exported to InfluxDB for visualisation in Grafana dashboards.

4.1 Data Preparation and Feature Engineering

As previously explained, the data consists of clickstream logs recorded by Matomo. To prepare the data set for clustering, several preprocessing and aggregation steps were necessary. These transformations converted raw, event-level data into structured, session-level feature vectors that capture relevant user behaviour dimensions suitable for clustering. Similar to previous research this includes session length, time spent on pages, and the number of products viewed [15, 34, 58].

The selection and structure of session-level features largely follow the approach established by Wagne and Neidhardt [53], who proposed a validated feature model for clustering user sessions on Geizhals. This manually engineered set of features resulted from iterative domain-expert consultations and validation through clustering experiments.

Our final set of features includes eleven session-level metrics grouped into three semantic categories: *Engagement*, *Browsing and Search*, and *Product Interest*. Table 4.1 summarises these features.

Wagne and Neidhardt used two additional features, `2ndLevelVisits` and `3rdLevelVisits`, which capture the number of visited second-level and third-level categories. These are omitted in our model, since they provided only marginal changes in the original analysis and did not materially affect the clustering outcome.

Table 4.1: Session features used for clustering

Feature	**Description**
Engagement	
npageViews	Total number of page views per session
timeSpent	Total duration of a session (in seconds)
meanPageviewTime	Mean dwell time per page (seconds), indicating session intensity
Browsing and Search	
nfilters	Total number of filter actions in category views
nsearches	Number of full-text search events
nddsearches	Number of interactions via dropdown search
dealActions	Number of interactions with the "Deals" section
Product Interest	
nproducts	Number of product detail pages viewed
nleads	Number of outbound clicks to retailer websites
productCompareActions	Number of product comparison interactions
avgLeadSum	Average price of products resulting in retailer leads

The features in our final model encapsulate behavioural signals such as search activity and purchase intent. Wagne and Neidhardt demonstrated that their feature set, including two additional category-level variables distinguishes six meaningful session types: "Quick peek", "Major purchase", "Constraint-based browsing", "Knowledge seeking", "Query & browse", and "Heavy browsing". These session types are described in detail in Section 2.3. Adopting their validated model with minor adjustments ensures methodological comparability and allows a clear focus on temporal behavioural evolution.

4.2 Clustering Pipeline

The entire clustering workflow is scripted in Python. Because the monthly Matomo exports are large CSV files, each file is stream-read in fixed chunks of 250,000 rows. This approach keeps memory usage predictable and enables on-the-fly preprocessing.

Within each chunk we keep only sessions originating from Austria and remove duplicates. The remaining event-level records are then aggregated up to the session level, producing the eleven features listed in Table 4.1. Sessions with fewer than three page views are discarded, as they generally lack sufficient behavioural information.

To address outliers, we computed the Z-score for each observation, that is, the number of

standard deviations it lies from the feature mean. We applied the rule to `npageViews`, `nproducts`, `timeSpent`, `nsearches` and `avgLeadSum`. Observations whose absolute Z-score exceeded four were removed. This procedure eliminates sessions left open in an idle browser tab as well as traffic from web crawlers, which are automated scripts that systematically fetch pages and can otherwise affect behavioural statistics.

The cleaned annual data set is used to fit a single `StandardScaler` [1](*master_scaler*) [41] and a Principal Component Analysis (PCA) model [2] with six components (*master_pca*) [21, 25]. Both objects are persisted and reused unchanged for every later time slice (monthly, weekday/weekend, or Black Friday). Applying the same scaler and PCA guarantees that all subsequent clusterings live in an identical six-dimensional coordinate system, so quality indices and feature values remain comparable across periods.

A `StandardScaler` centres every numeric feature at zero and rescales it to unit variance. This step puts page-view counts and time values onto a common scale, so no single dimension can dominate the Euclidean distances that K-Means relies on. PCA is then applied to the standardised variables. It rotates them into a new set of orthogonal axes ordered by explained variance. This highlights meaningful variance and improves cluster cohesion. With this we reduced the feature space from eleven to six dimensions. We selected six components for the PCA based on Wagne and Neidhardt's finding that this number yields best results.

Clustering is then carried out with the Mini-Batch K-Means algorithm. Mini-Batch K-means is a variant of the traditional K-Means clustering algorithm explained in Section 2.4 designed to handle large data sets more efficiently [47, 46, 49]. Instead of processing the entire data set in each iteration, it uses small subsets of the data, called batches, which significantly reduces computational time and memory usage. Further optimization involves the use of the k-means++ technique [3]. Instead of selecting all initial centroids randomly it strategically chooses them one by one, ensuring they are well-spread across the data set. This improves convergence speed.

Following Wagne and Neidhardt, we set the mini-batch size to 100,000 sessions. Their study suggested that six clusters capture the dominant behavioural patterns. We validated this choice by computing cluster quality metrics on our full data set, including the Sum of Squared Errors (SSE), Silhouette Score [43], and Calinski-Harabasz Index [8]. As shown in Table 4.2, all three metrics confirmed that $k = 6$ is also appropriate for our data set.

These three validity indices each capture a different aspect of cluster quality. The SSE measures the total squared distance between each session and the center of its assigned cluster. Lower SSE values indicate more compact and cohesive groups. The Silhouette Score evaluates how similar a session is to its own cluster compared to the next closest cluster, with scores ranging from -1 to 1. Higher scores suggest better separation and

[1] `https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html`

[2] `https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html`

Table 4.2: Cluster quality metrics for different values of k on the yearly data set.

k	**Silhouette Score**	**SSE**	**Calinski-Harabasz Index**
2	0.461	49,520,104	1,727,572
3	0.415	42,151,510	1,630,128
4	0.410	38,257,534	1,436,199
5	0.414	33,881,166	1,443,021
6	0.410	31,391,689	1,358,780
7	0.377	28,375,742	1,384,236
8	0.183	28,892,594	1,143,990
9	0.279	24,520,174	1,334,653
10	0.239	23,731,828	1,253,402

clearer structure. Lastly, the Calinski-Harabasz Index compares the distance between cluster centers to the spread within clusters. A higher value implies that clusters are both internally compact and well separated from one another.

With this, we defined the clustering pipeline. The next section describes how we apply it to examine how behavioural patterns evolve.

4.3 Temporal Clustering

To answer the three research questions stated in Section 1.2 we filter the data by time stamp, transform the resulting slice with the stored scaler (`master_scaler`) and PCA (`master_pca`), and run Mini-Batch K-Means with the fixed hyper-parameters from Section 4.2. So the clustering is performed separately for each period.

As previously mentioned, the global scaler and PCA trained on the full 2024-25 data set allow every filtered data set to be projected into the same six-dimensional space.

Month-to-month. To address **RQ1** we cluster the sessions of each calendar month between July 2024 and June 2025 separately, yielding twelve solutions that can be compared for seasonal shifts.

Weekday versus weekend. To address **RQ2** we partition the year's data into a weekday subset (Monday-Thursday) and a weekend subset (Saturday-Sunday). Fridays were excluded due to their hybrid consumer behaviour patterns [13]. Each subset is clustered independently and then compared.

Black Friday window. For **RQ3**, we isolate the extended Black Friday period from 21 November 2024 to 2 December 2024, covering twelve consecutive days. This period was chosen due to the elevated shopping activity as seen in Figure 4.1. During this

interval, the average daily sessions are 42,232, compared to the combined November-December average of 26,614. We cluster this interval separately and compare it against the remaining 12-month sample, excluding the detected Black Friday period itself.

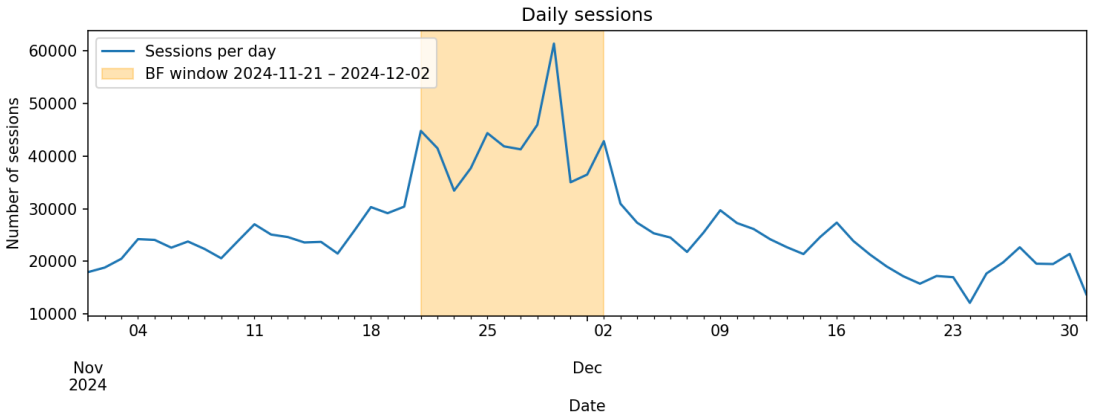


Figure 4.1: Daily sessions in November-December 2024 with the Black Friday window highlighted (21 November-2 December 2024)

4.3.1 Cluster-Label Consistency

Because the pipeline produces one clustering for every time slice, the integer labels returned by Mini-Batch K-Means cannot be compared directly: label 0 in July is not necessarily the same behavioural group as label 0 in August, since label numbers are assigned arbitrarily. To stabilise the cluster labels we first cluster the full 12-month data set and store its six centroids as a reference set. After each subsequent clustering we align the new centroids with this reference by solving a linear assignment problem using the Hungarian algorithm [28], which minimises the total pairwise Euclidean distance. The resulting permutation is then applied to the predicted labels, ensuring that every behavioural cluster carries a time-stable identifier across all experiments.

4.3.2 Metrics Exported to InfluxDB

After every clustering run we store a set of metrics in an InfluxDB bucket, which makes the results queryable and ready for visualisation in Grafana. First, we record the Silhouette Score, which indicates how well separated and cohesive the resulting clusters are. Alongside this score, we store the absolute and relative size of each cluster, because changes in membership proportions are an important signal of temporal behavioural shifts.

For feature-level interpretation we follow a two-step process. First, we calculate the average value of each session feature within every cluster. This results in a set of feature averages for each of the six clusters. Second, we standardise these values using a scaler

trained on the cluster-level feature means of the full-year baseline. This places all features on the same scale with zero mean and unit variance, making them comparable across different time periods. The resulting standardised feature profiles are stored and form the basis for the cluster heat maps discussed in Chapter 5.

In addition to descriptive metrics, statistical tests are applied to assess whether observed differences in cluster distributions between specific temporal baselines are statistically significant. For each behavioural cluster, a two-sample proportion z-test is performed to compare the proportion of sessions assigned to that cluster across the two baselines. The null hypothesis H_0 assumes equal proportions in both baselines, while the alternative hypothesis H_A tests for any difference. P-values below 0.01 are considered statistically significant.

The metrics introduced in this chapter are stored in InfluxDB and provide the basis for the Grafana panels and the result analysis presented in the following chapter.

CHAPTER 5

Results and Discussion

This chapter reports the outcomes of the temporal clustering pipeline described in Chapter 4. We begin with the year-wide solution and assess how closely it reproduces the six session session types identified by Wagne and Neidhardt [53]. The remainder of the chapter turns to the three research questions. For each temporal slice we first summarise cluster-quality indices, then compare the cluster distribution with the appropriate baseline combining visual inspection with statistical tests. Finally we examine behavioural shifts in greater depth through standardised heat maps of feature means. All figures are exported from a Grafana dashboard that visualises the metrics stored in InfluxDB.

5.1 Year-Wide Baseline

Before analysing individual time slices, we first clustered the complete data set. This serves two purposes: first, to establish a global behavioural baseline, and second, to enable comparison with the findings of Wagne and Neidhardt based on the earlier 2023-24 data set.

5.1.1 Clustering Quality

The year-wide solution achieves a silhouette coefficient of 0.41, closely matching the 0.37 reported by Wagne and Neidhardt. This indicates a partition of solid but not perfect quality. Some overlap remains, which is expected in noisy clickstream data [1].

5.1.2 Cluster Composition

The resulting cluster distribution is visualised in Figure 5.1. Cluster 0 dominates with 64 % of all sessions, whereas Cluster 5 is the smallest with just over 2 %. The ranking and proportions replicate the pattern observed by Wagne and Neidhardt almost exactly, which confirms that the underlying sessions have remained stable from 2023-24 to 2024-25.

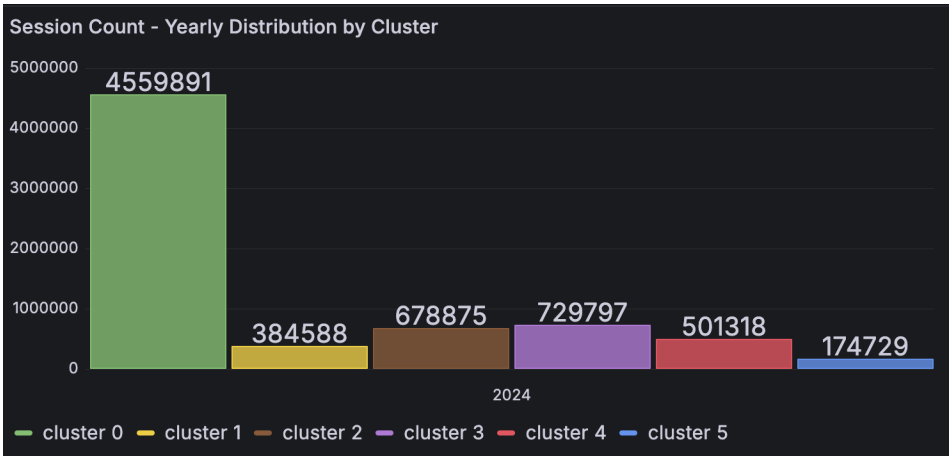


Figure 5.1: Absolute session distribution across clusters for the full data set

5.1.3 Feature Profiles

Scaled Feature Means - Yearly Cluster Heatmap

0	1	2	3	4	5	overall	feature
-0.50	2.23	-0.48	-0.39	-0.44	-0.41	-0.33	avgLeadSum
-0.45	-0.56	-0.12	-0.46	-0.61	2.21	-0.37	dealActions
-0.73	-0.02	-0.56	2.18	-0.40	-0.48	-0.34	meanPageviewTime
-0.56	-0.34	-0.83	-0.27	2.18	-0.18	-0.34	nddsearches
-0.64	-0.61	0.61	-0.66	-0.69	1.98	-0.46	nfilters
-1.01	1.97	-0.66	-0.58	-0.25	0.53	-0.67	nleads
-0.81	-0.67	0.24	-0.76	-0.06	2.06	-0.57	npageViews
-0.89	-0.70	0.24	-0.72	0.04	2.03	-0.62	nproducts
-0.42	-0.39	-0.70	-0.47	2.21	-0.23	-0.26	nsearches
-0.56	-0.53	-0.13	-0.51	-0.49	2.21	-0.44	productCompareActions
-1.39	-0.64	-0.30	0.88	-0.21	1.66	-0.85	timeSpent

Figure 5.2: Standardised feature means for the six clusters in the full data set

Figure 5.2 visualises standardised feature means and confirms that the clusters mostly replicate the session types introduced in Section 2.3.

Cluster 0 aligns with the “Quick peek” type, showing very short visits without retailer leads. Cluster 1 matches the “Major purchase” pattern and generates outbound leads. Cluster 2 shows some similarities to the “Constraint-based browsing” type, particularly in

its moderate use of filters. However, the feature profile is less pronounced in this regard. Compared to the other clusters in our model, filter usage is only slightly above average and does not dominate the cluster's behavioural signature. Another difference is that product comparison actions are underrepresented relative to the other clusters, which contrasts with the reference model, where such actions were a defining characteristic. Cluster 3 corresponds to the "Knowledge seeking" profile, characterised by long dwell times on product pages, while Cluster 4 aligns with the "Query & browse" and performs the most searches on the site. Cluster 5 has elevated values on almost every dimension, matching the "Heavy browsing" group.

These six profiles serve as the reference frame for the remainder of the chapter. The following sections trace how their size and behavioural signature evolve across months, between weekdays and weekends, and during the Black Friday window. Because five clusters replicate the session types of Wagne and Neidhardt, we keep their original labels. Cluster 2, which diverges from the reference is referred to simply by its numeric identifier.

5.2 RQ1 - Month-to-Month Dynamics

This section analyses how user behaviour changes from month to month. We begin by examining the overall traffic volume and how sessions are distributed across clusters over time. Next, we assess the clustering quality for each month to evaluate the stability of the behavioural structure. Finally, we study the feature heat maps to identify behavioural shifts in more detail, combining them with silhouette scores and cluster distributions to trace temporal trends.

5.2.1 Monthly Traffic Volume

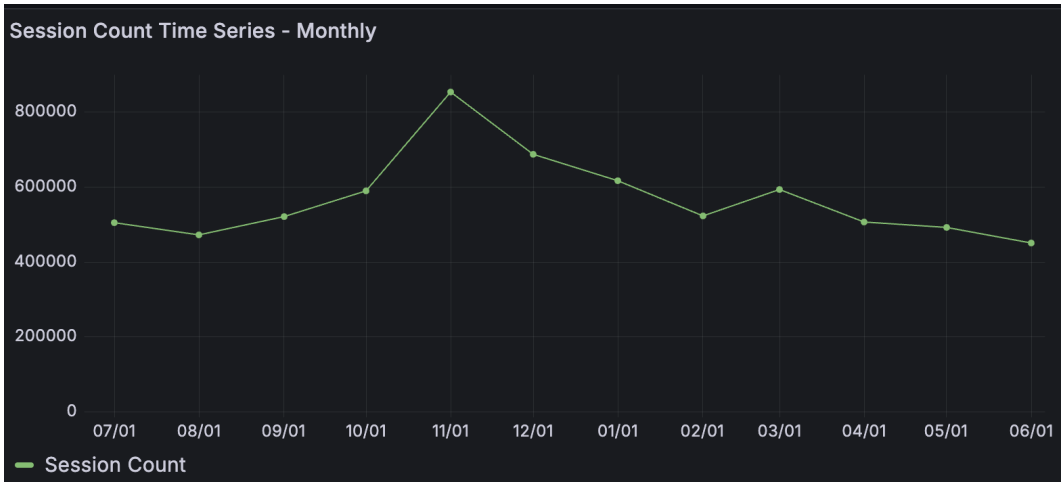


Figure 5.3: Monthly session count from July 2024 to June 2025

Figure 5.3 shows that monthly traffic fluctuates between approximately 460,000 and 860,000 sessions, based on the filtered dataset. Demand softens during the summer, rises through September and October, and reaches its peak in November. Traffic then recedes in January and stabilises around half a million sessions for the remainder of the year.

5.2.2 Clustering Quality

As shown in Table 5.1, the silhouette scores vary considerably across months. In eight out of twelve months, the score exceeds 0.30, indicating reasonably well-separated clusters. The lowest values are observed in January ($s = 0.194$) and May ($s = 0.232$), while July and October lie at the borderline with identical scores of $s = 0.279$.

Table 5.1: Monthly silhouette scores

Month	Silhouette Score
2024-07	0.279
2024-08	0.399
2024-09	0.439
2024-10	0.383
2024-11	0.279
2024-12	0.424
2025-01	0.194
2025-02	0.421
2025-03	0.410
2025-04	0.397
2025-05	0.232
2025-06	0.302

Overall, the monthly clusterings can be considered reliable, but interpretations for months with lower silhouette scores should be made with caution due to weaker separation.

5.2.3 Cluster-Size Evolution

Figure 5.4 shows how the absolute session count per cluster changes over the year. Two distinct patterns can be observed:

1. Quick peek dominates throughout the year. "Quick peek" (Cluster 0) remains the most common cluster in every month, ranging from 47.7 % in January to 69.7 % in September. This indicates that quick price checks make up the majority of sessions, regardless of the time of year.

2. Buying intent peaks in November. "Major purchase" (Cluster 1) remains relatively stable across most months but rises sharply in November, suggesting that this month represents a peak period for purchases of more expensive products.

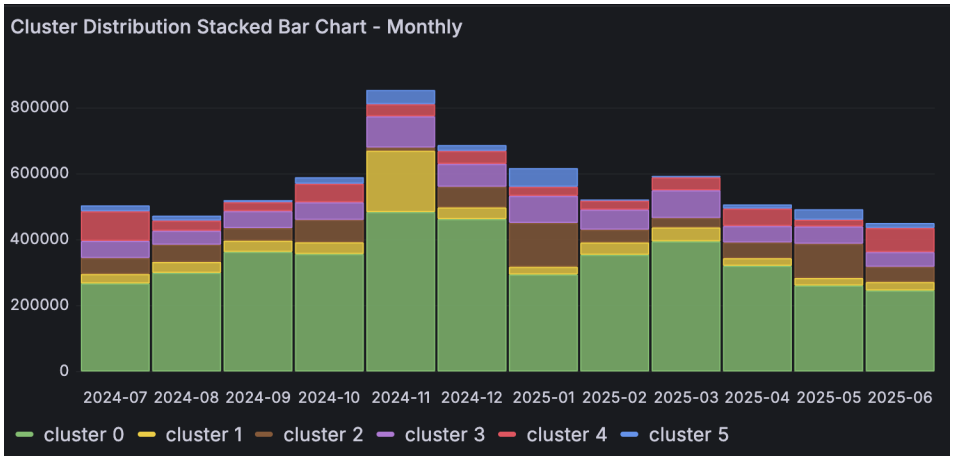


Figure 5.4: Monthly session distribution across clusters from July 2024 to June 2025

Compared to the yearly distribution, the months of July, October, January, and May show a noticeably different structure. These are also the months with a low silhouette score, suggesting a potential link between lower cluster separation and the observed distribution shifts.

To determine whether the observed changes in cluster proportions are statistically significant, we perform two-sample proportion z-tests between each month and the full-sample baseline. Figure 5.5 summarises the results as a heat map, where cell colours encode the absolute deviation in cluster share, and annotations report both the percentage change and the corresponding z-score.

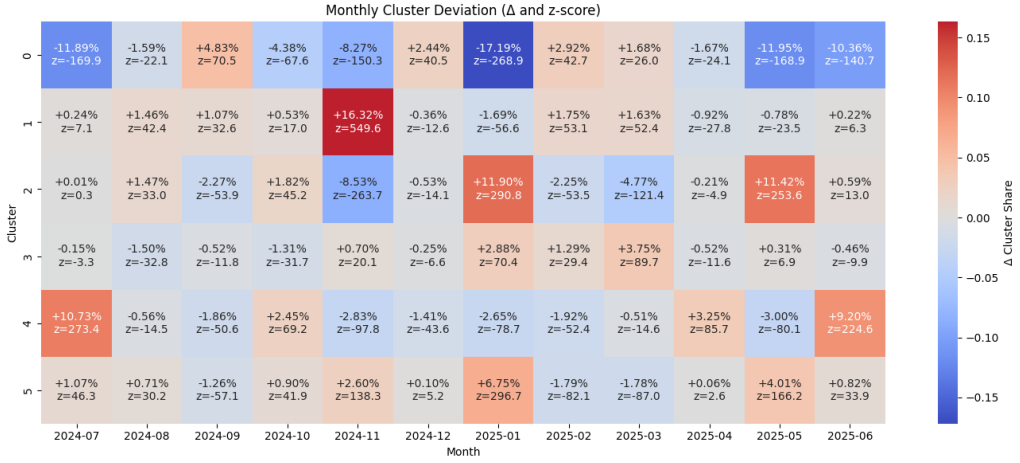


Figure 5.5: Statistical comparison of monthly cluster proportions against the full sample

The statistical analysis confirms that the majority of month-to-month shifts are highly significant. Most deviations yield p-values below 0.001, providing strong evidence against

the null hypothesis of equal cluster proportions.

Notably, the months of July, October, January, and May stand out with the largest deviations across multiple clusters, confirming the visual trends.

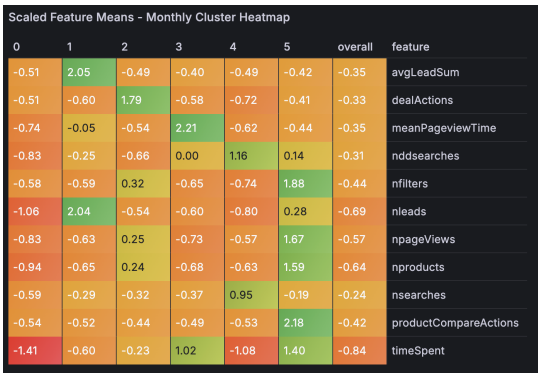
Cluster stability also varies across behavioural types. "Knowledge seeking" (Cluster 3) remains stable, exhibiting fewer deviations than other clusters. In contrast, "Quick peek" (Cluster 0), "Major purchase" (Cluster 1), and Cluster 2 show the most fluctuations. This pattern suggests that impulsive lookups, purchase-oriented sessions, and deal-seeking behaviours are particularly sensitive to external and seasonal factors.

One of the most pronounced shifts occurs in November. The share of "Major purchase" sessions increases by 16.32% relative to the annual baseline, while Cluster 2 and "Quick peek" shrink significantly. This indicates a behavioural transition from exploratory browsing to targeted purchasing during the peak shopping season.

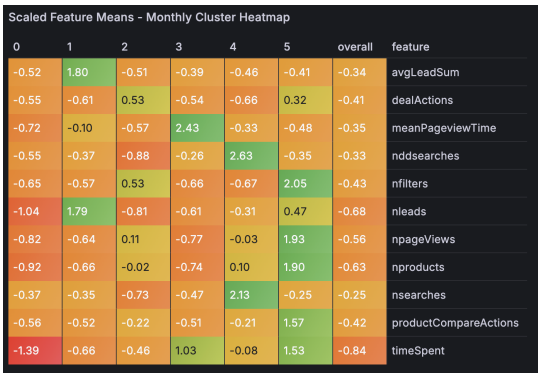
These results support the patterns seen in Figure 5.4 and provide statistical evidence for seasonal changes in user behaviour.

5.2.4 Feature Profiles

To understand how user behaviour shifts over time, we analyse the monthly feature heat maps. The analysis proceeds chronologically, grouping months with similar contexts to reveal broader seasonal patterns. We interpret the heat maps in combination with silhouette scores and cluster distributions to capture both structural and behavioural changes.



(a) July 2024 heat map

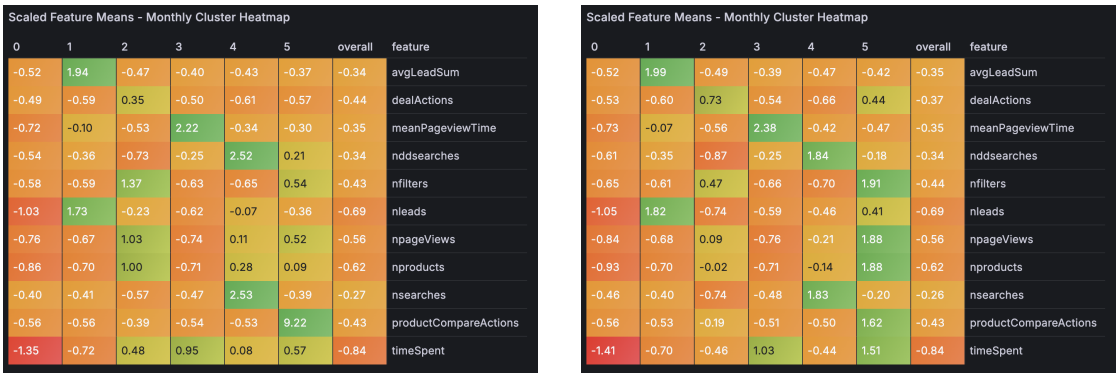


(b) August 2024 heat map

Figure 5.6: Standardised feature means for each cluster in July and August 2024

Summer lull: July and August (Figure 5.6). Traffic is at its seasonal low with silhouette scores of 0.28 and 0.4. "Quick peek" still occupies around two thirds of all sessions. July shows the first clear hint of deal seeking. Compared to the annual reference, where `dealActions` are concentrated in the "Heavy browsing" (Cluster 5), the July

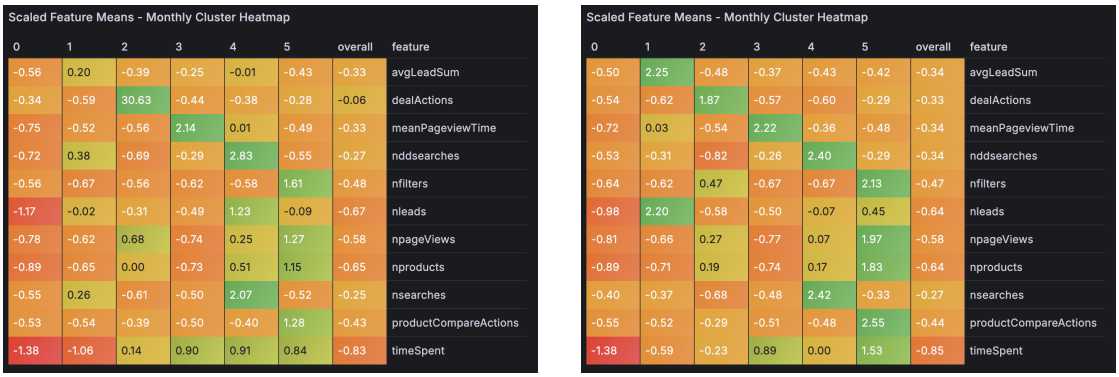
heat map shows this spike shifted to Cluster 2. In August this spike shrinks again and the heat map resembles the annual reference almost perfectly.



(a) September 2024 heat map (b) October 2024 heat map

Figure 5.7: Standardised feature means for each cluster in September and October 2024

Early autumn build-up: September and October (Figure 5.7). Both months have silhouette score above 0.38 and follow the baseline distribution. September amplifies filter use and product views inside Cluster 2, signalling slow preparation for the shopping season, while browsing intensity in "Heavy browsing" declines and its members shift part of their comparison activity to Cluster 2. October completes that migration: Cluster 2 becomes the dedicated deal-hunter segment, the "Heavy browsing" cluster loses deal traffic entirely, but the overall mix is still balanced.



(a) November 2024 heat map (b) December 2024 heat map

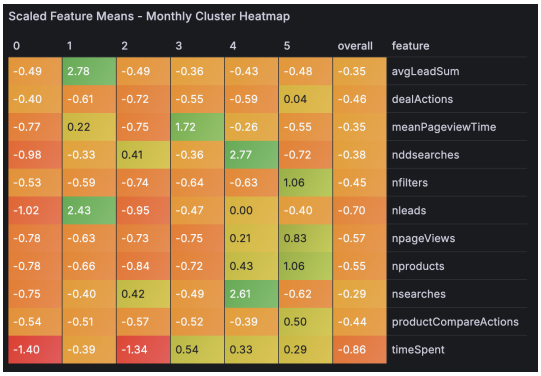
Figure 5.8: Standardised feature means for each cluster in November and December 2024

Peak season and immediate aftermath: November to January (Figures 5.8 and 5.9a). November traffic increases and the silhouette drops to 0.28, indicating overlap between clusters. "Major purchase" more than quadruples in share, but its lead count falls and search volume rises, suggesting that buyers are comparing offers carefully

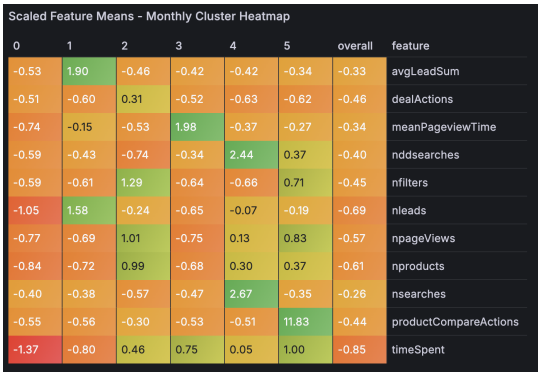
before making a decision. Cluster 2 shows a drastic spike in `dealActions`, indicating highly focused deal-seeking behaviour, while Cluster 4 absorbs the heavier browsing behaviour.

December returns to the year-wide proportions with a silhouette of 0.42, though Cluster 2 still captures most deal clicks.

In January behavioural boundaries blur ($s = 0.19$). Cluster 2 and "Query & browse" (Cluster 4) increase, each marked by elevated search intensity but little dwell time. "Heavy browsing" loses structure. Browsing activity declines and deal interactions disappear. This pattern suggests that users are checking prices after the holidays rather than making serious purchase decisions.

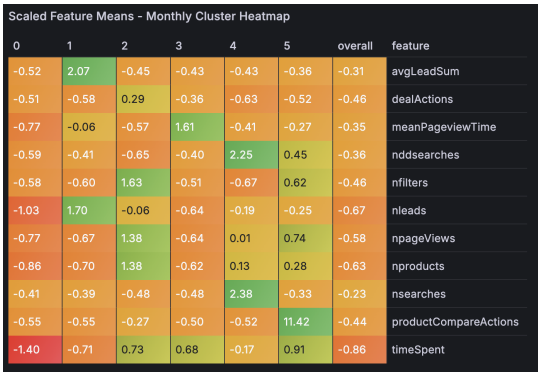


(a) January 2025 heat map



(b) February 2025 heat map

Figure 5.9: Standardised feature means for each cluster in January and February 2025



(a) March 2025 heat map

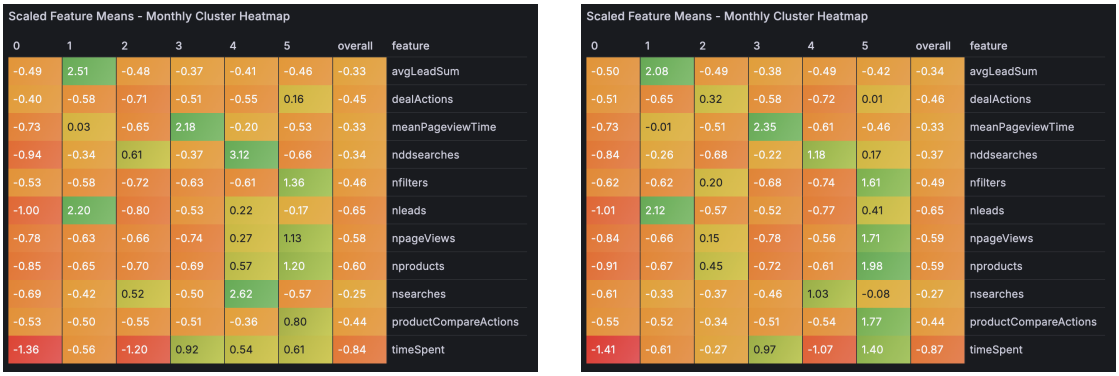
(b) April 2025 heat map

Figure 5.10: Standardised feature means for each cluster in March and April 2025

Stabilisation and rebalancing: February to April (Figures 5.9b and 5.10). The silhouette rebounds above 0.40 in February and March. Distribution and features realign with the baseline, except that "Heavy browsing" continues to lose deal traffic and

is now dominated by product comparisons instead. Cluster 2 keeps a mild deal focus and exhibits generally stronger browsing than in autumn, indicating that users compare more broadly.

April is nearly identical to the annual profile except for the absence of deal interactions in "Heavy browsing".



(a) May 2025 heat map

(b) June 2025 heat map

Figure 5.11: Standardised feature means for each cluster in May and June 2025

Late-spring anomaly in May and stabilisation in June. In May the silhouette dips again ($s = 0.23$) and the distribution deviates from the baseline. Cluster 2 doubles in size, and together with "Query & browse" they both show strong search signals while lead generation remains modest. "Heavy browsing" grows in proportion, but shows weaker values across all browsing features. In June, behaviour returns to the year-wide baseline. "Query & browse" shows slightly elevated search activity, while "Heavy browsing" still interacts little with the deals section.

Throughout the year, we notice clear seasonal patterns in cluster behaviour. Months with high silhouette scores, such as August, September, February, and March closely follow the year-wide baseline. In these months user behaviour stays consistent. In contrast, months with lower scores like November, January, and May show stronger changes in how users behave.

Overall, the analysis shows that seasonal events affect not only how many people visit the site, but also how they use it. We often see patterns such as focused deal-seeking, increased searching, or more scattered behaviour after promotions. These short-term effects can influence the otherwise mostly stable structure of behavioural clusters.

5.3 RQ2 - Weekday versus Weekend Differences

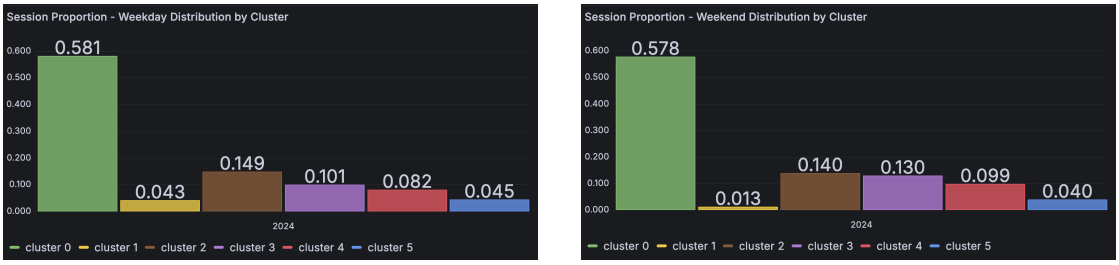
This section investigates differences between weekdays (Monday through Thursday) and weekends (Saturday and Sunday). We first assess cluster quality metrics for both periods,

then examine the cluster distribution and finally explore how behavioural profiles differ across clusters.

5.3.1 Clustering Quality

The weekday clustering yields a silhouette coefficient of 0.34. In comparison, the weekend clustering shows slightly lower quality with a silhouette coefficient of 0.31. Despite the drop, the metrics remain within the acceptable range and therefore support interpretation.

5.3.2 Cluster Composition



(a) Weekday distribution

(b) Weekend distribution

Figure 5.12: Relative session distribution across clusters for weekdays and weekends

Figure 5.12 shows that the overall distribution is stable across the week: "Quick peek" (Cluster 0) remains by far the largest category, slipping only from 58.1 % on workdays to 57.8 % at weekends. This 0.3 % change is statistically detectable ($z = 7.6$, $p < 0.001$) but negligible in practice and confirms that casual price checks are the most common behaviour on both weekdays and weekends.

Purchase-driven traffic behaves very differently. "Major purchase" (Cluster 1) falls from 4.3 % to 1.3 %, a relative decline of almost 70 % ($z = 185$, $p < 10^{-15}$). Feature profiles in the next subsection show that these sessions are characterised by high lead counts. This weekday bias suggests that more users continue to retailer sites during workdays.

Exploratory patterns become more common in the space left by fewer purchase sessions. "Knowledge seeking" (Cluster 3) rises from 10.1 % to 13.0 % and "Query & browse" (Cluster 4) from 8.2 % to 9.9 %. Both shifts are strongly significant ($z = -107.6$ and $z = -68.5$, $p < 10^{-15}$) which is consistent with the interpretation that weekend visitors tend to read more product pages and submit more queries.

The remaining Cluster 2 and "Heavy browsing" (Cluster 5) change only slightly, which matches their moderate level of activity.

In summary we can see that weekends suppress purchase-driven behaviour and favour more exploratory or research-oriented browsing. The pattern supports findings from earlier studies, which report more exploratory browsing and less purchase-oriented activity at weekends [20].

5.3.3 Feature Profiles

Scaled Feature Means - Weekday Cluster Heatmap

0	1	2	3	4	5	overall	feature
-0.49	2.64	-0.49	-0.38	-0.43	-0.44	-0.34	avgLeadSum
-0.62	-0.59	0.77	-0.55	-0.64	0.52	-0.36	dealActions
-0.73	0.05	-0.61	2.28	-0.36	-0.47	-0.34	meanPageviewTime
-0.53	-0.35	-0.91	-0.29	2.08	-0.38	-0.34	nddsearches
-0.69	-0.60	0.14	-0.66	-0.68	1.65	-0.46	nfilters
-1.01	2.19	-0.83	-0.57	-0.25	0.12	-0.69	nleads
-0.86	-0.66	-0.19	-0.77	-0.11	1.53	-0.57	npageViews
-0.94	-0.69	-0.26	-0.72	0.01	1.51	-0.62	nproducts
-0.38	-0.39	-0.77	-0.48	2.00	-0.35	-0.26	nsearches
-0.57	-0.52	-0.32	-0.52	-0.49	1.33	-0.44	productCompareActions
-1.42	-0.56	-0.79	0.94	-0.22	1.16	-0.84	timeSpent

(a) Weekday heat map

Scaled Feature Means - Weekend Cluster Heatmap

0	1	2	3	4	5	overall	feature
-0.48	0.06	-0.49	0.57	-0.44	-0.44	-0.33	avgLeadSum
-0.55	10.90	-0.59	-0.65	-0.69	-0.62	-0.44	dealActions
-0.73	-0.26	-0.58	1.62	-0.45	-0.47	-0.36	meanPageviewTime
-0.57	1.42	-0.93	-0.34	1.49	-0.45	-0.36	nddsearches
-0.70	-0.42	0.22	-0.66	-0.69	1.86	-0.46	nfilters
-1.00	2.82	-0.83	0.44	-0.43	0.00	-0.64	nleads
-0.86	0.85	-0.17	-0.77	-0.22	1.59	-0.57	npageViews
-0.95	1.03	-0.13	-0.76	-0.22	1.56	-0.61	nproducts
-0.44	0.84	-0.80	-0.52	1.83	-0.44	-0.26	nsearches
-0.58	-0.45	-0.33	-0.53	-0.51	1.72	-0.44	productCompareActions
-1.43	1.20	-0.74	0.41	-0.52	1.18	-0.86	timeSpent

(b) Weekend heat map

Figure 5.13: Standardised feature means for each cluster on weekdays and weekends

Figure 5.13 shows the standardised mean values of the eleven session features for each cluster, separately for weekdays and weekends.

During the working week the six clusters reproduce the patterns already described for the year-wide baseline: “Quick peek” is low on every engagement metric, “Major purchase” excels in `avgLeadSum` and `nleads`, and “Knowledge seeking” shows the highest dwell time. The other clusters also correspond to the year-wide baseline.

At weekends these signatures remain recognisable, yet several dimensions intensify or soften in a way that clarifies the behavioural shift suggested by the distribution analysis.

One of the most intense deviation is in the “Major purchase” cluster. Here behaviour changes as the `dealAction` feature explodes from a small negative deviation to an extreme positive. This indicates a lot of interaction with the "Deals" section that contributes to the decrease in the `avgLeadSum` feature. The number of leads generated is comparable to the weekday group.

“Knowledge seeking” shows a slight decrease in the page-view dwell time, but now records positive deviations in both `avgLeadSum` and `nleads`. Weekend researchers therefore appear more willing to follow through to a retailer once they have gathered sufficient information.

The behaviour on the other clusters seems similar between weekdays and weekends.

Across clusters the overall level of the features changes only marginally. In combination, the heat maps and the proportion analysis show a coherent story. Weekends attract fewer decisive buyers but encourage more extensive browsing and comparison, while the remaining buyers concentrate on bargain hunting.

The comparison between weekdays and weekends reveals stable overall behaviour, but also clear shifts in user intent. Weekdays are marked by more decisive browsing patterns,

where users tend to continue to retailer sites. Weekends show a stronger orientation towards exploration and information gathering. Users are more likely to browse, compare products, and engage with promotional sections of the site. The overall structure of user behaviour stays mostly the same, but small changes in feature values show that users use the platform differently depending on the day. This supports the idea that shopping behaviour is influenced not only by personal needs but also by the time, like workdays versus weekends.

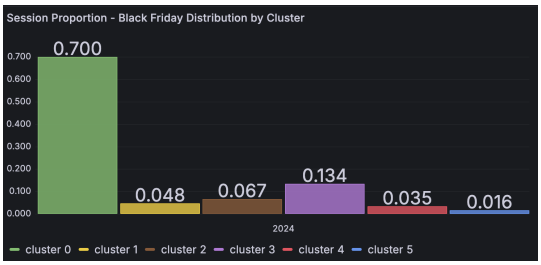
5.4 RQ3 - Black Friday Behaviour

This section examines how user behaviour changes during the twelve day Black Friday window from 21 November to 2 December 2024. We first discuss clustering quality, then compare the cluster distribution with the non-Black Friday baseline, defined as the remaining 12-month sample excluding this interval, and finally interpret the feature profiles.

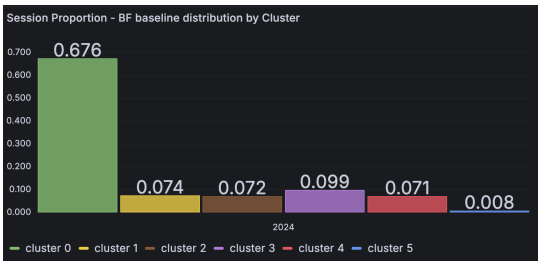
5.4.1 Clustering Quality

The silhouette score for the Black Friday interval is 0.40, which is almost identical to the non-Black Friday baseline with 0.42. This indicates clear separation despite the sharp increase in traffic. The clustering structure can therefore be considered sufficiently stable to support interpretation.

5.4.2 Cluster Composition



(a) Black Friday distribution



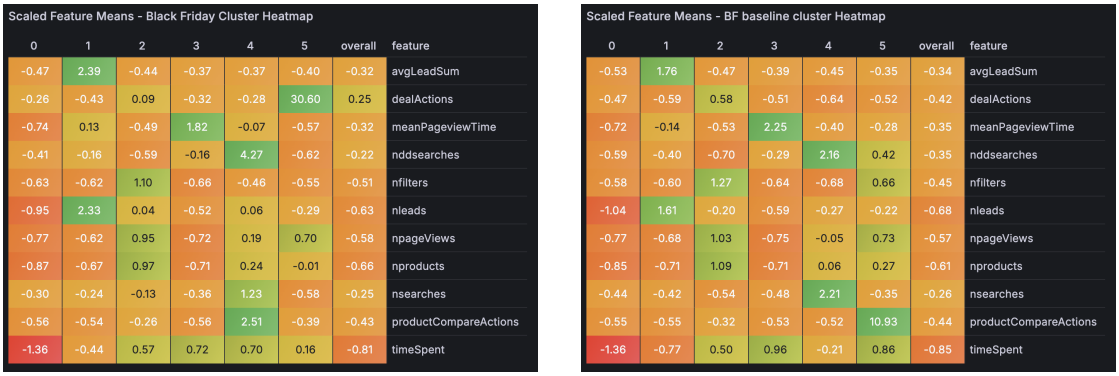
(b) non-Black Friday baseline distribution

Figure 5.14: Relative session distribution across clusters for Black Friday and the corresponding baseline

Figure 5.14 contrasts the Black Friday distribution with the non-Black Friday baseline that excludes the event itself. The overall composition shifts in a distinct way. "Quick peek" (Cluster 0) grows from 67.6 % in the baseline to 70.0 % during Black Friday ($z = 35.6$, $p < 0.001$), showing that very short price checks become even more common in the campaign period. In contrast, "Major purchase" (Cluster 1) declines from 7.4 % to 4.8 % ($z = -69.1$, $p < 0.001$), although it still represents the cluster with the strongest purchase signals. Cluster 2 remains almost unchanged, moving only slightly from 7.2 %

to 6.7 % ($z = -12.6$, $p < 0.001$). More pronounced shifts are visible in the other groups. "Knowledge seeking" (Cluster 3) rises from 9.9 % to 13.4 % ($z = 78.6$, $p < 0.001$), whereas "Query & browse" (Cluster 4) drops sharply from 7.1 % to 3.5 % ($z = -98.4$, $p < 0.001$). Finally, "Heavy browsing" (Cluster 5) doubles from 0.8 % to 1.6 % ($z = 60.6$, $p < 0.001$). Taken together, these results show that the Black Friday event polarises activity. Many sessions reduce to brief check-ins, while others shift toward careful reading and deal-oriented browsing.

5.4.3 Feature Profiles



(a) Black Friday heat map (b) non-Black Friday baseline heat map

Figure 5.15: Standardised feature means for each cluster during Black Friday and the corresponding baseline

Figure 5.15 compares the standardised feature means during the Black Friday period with the non-Black Friday baseline. While most clusters retain their familiar signatures, several dimensions intensify in ways that help explain the distributional shifts reported above.

"Quick peek", "Major purchase", Cluster 2, and "Knowledge seeking" remain largely consistent with their year-wide profiles. Their feature patterns differ only slightly between the Black Friday window and the baseline, which underlines their structural stability.

In contrast, "Query & browse" becomes less common, but shows stronger activity across several browsing dimensions. In particular, `productCompareActions` rises sharply, suggesting that sessions that use product comparisons are partly absorbed into this cluster during the campaign period.

The most pronounced shift occurs in "Heavy browsing". In the baseline this group was characterised mainly by elevated product comparisons, but during Black Friday it develops into the cluster most strongly associated with interactions with the deals section. The mean value of the `dealActions` feature rises sharply, turning this cluster into the primary point of interaction with deals section.

Overall, the Black Friday event does not simply increase purchase-driven sessions. Instead, behaviour becomes polarised. Many users restrict their activity to quick price checks, while a smaller subgroup engages in more intensive research and deal-focused exploration. At the same time, exploratory browsing through "Query & browse" declines, and interactions with promotions concentrate in a single cluster.

These results complete the analysis of temporal behavioural patterns, covering month-to-month dynamics, weekday-weekend differences, and behavioural shifts during the Black Friday period. In the following chapter, we revisit the research questions and summarise the key findings. We also highlight limitations of the current approach and outline possible directions for future research.

CHAPTER 6

Conclusion and Future Work

This thesis investigated temporal and contextual variations in user behaviour on a price-comparison platform by applying unsupervised clustering to one year of clickstream sessions from Geizhals, covering July 2024 to June 2025. Three research questions guided the analysis: month-to-month dynamics (RQ1), weekday-weekend differences (RQ2), and behavioural shifts during the Black Friday period (RQ3). Extending previous work on static session segmentation, the study introduced a temporal clustering pipeline that enabled comparison of behavioural patterns across time. Cluster stability was ensured through centroid alignment, and the results were examined using statistical tests as well as visualisations in the form of bar plots and feature heat maps.

To establish a global behavioural baseline, we first clustered the complete data set from July 2024 to June 2025. This year-wide solution serves two purposes: it enables direct comparison with the static segmentation reported by Wagne and Neidhardt [53], and it provides a reference frame for the temporal analyses in the following sections. The results confirm that session behaviour has remained remarkably stable: five of the six clusters closely reproduce the session types described in Section 2.3, while Cluster 2 shows some divergence. This consistency validates both the feature model and the temporal clustering pipeline, and allows the subsequent month-to-month, weekday-weekend, and Black Friday analyses to be interpreted against a reliable baseline.

For **RQ1**, the month-to-month analysis revealed that "Quick peek" sessions dominate throughout the year, confirming that the majority of visits are brief price or availability checks. The overall cluster structure remains stable for much of the year and closely resembles the year-wide baseline. However, distinct seasonal variations emerged. July and August reflect the summer lull with reduced traffic. September and October show a gradual build-up towards the shopping season. November stands out with a marked increase in purchase-oriented sessions. January exhibits diffuse behaviour with blurred cluster boundaries, while February and March show a rapid return to the baseline structure. April and June are nearly identical to the annual profile, whereas May deviates

with elevated search activity and weaker purchase signals. These findings suggest that peak shopping periods and their aftermath strongly influence both the volume and the behavioural composition of sessions, while other months remain largely stable.

For **RQ2**, the comparison between weekdays and weekends showed that the overall cluster structure remains stable. However, user intent shifts noticeably. Purchase-oriented sessions decline at weekends, while exploratory patterns become more common. Feature profiles confirm this interpretation. Weekday sessions generate more outbound leads, whereas weekend sessions involve more extensive product browsing, queries, and interactions with promotional sections. These findings suggest that workdays encourage more decisive purchasing behaviour, while weekends are characterised by exploration and information gathering.

For **RQ3**, the Black Friday analysis showed that the event did not simply increase purchase-oriented behaviour but polarised it. Traffic rose sharply, with quick check-ins becoming more frequent and knowledge-seeking behaviour also gaining importance. As expected, interactions with the deals section shifted markedly, as many sessions concentrated on promotions. These findings suggest that Black Friday does not simply amplify demand but changes how users engage. It reinforces common behaviours such as quick visits and knowledge seeking, while directing more specialised actions like deal exploration into clear patterns.

Taken together, these results show that while the dominant behavioural segments remain stable over time, their relative prevalence and feature composition shift in response to seasonal, weekly, and promotional contexts. The findings highlight the value of temporal clustering for uncovering such dynamics and provide a methodological framework for monitoring behavioural change on price-comparison platforms.

Furthermore, the developed Grafana dashboard represents a significant practical contribution of this work. By integrating the clustering results into an interactive monitoring interface, the thesis provides a functional framework for the observation and interpretation of session distributions. This tool enables the immediate detection of behavioural shifts during future promotional events, making complex analytical insights accessible to non-technical stakeholders and supporting data-driven platform strategies.

6.1 Limitations

Several limitations should be acknowledged when interpreting the results of this study. First, the analysis was restricted to a single price-comparison platform, a single year of data, and sessions originating exclusively from the Austrian market. This specific geographic, platform-specific, and temporal scope may limit the generalisability of the findings to other services, different regional markets (such as the UK or Poland), or behavioural trends observed over much longer periods. Furthermore, the temporal segmentation of the data assumes that user behaviour remains sufficiently homogeneous within each defined time slice. However, periods with overlapping or rapidly changing

behaviour—such as the Black Friday promotional window or the post-holiday shift in January—may reduce cluster separation and challenge this assumption. This is reflected in the fluctuating silhouette scores, where months with higher volatility exhibit weaker separation between the identified session types.

In addition, limitations arise from preprocessing choices. The definition of sessions, the filtering of very short or inactive sessions, and the aggregation of actions into broader feature categories may all have influenced the observed patterns. While these steps were necessary for tractable analysis, they could also have introduced biases that affect the stability and interpretation of the results.

Another limitation is that the clustering was performed at the session level. On the web, consistent cross-session user tracking is inherently unreliable due to privacy regulations, cookie lifetimes, cross-device usage, and browser tracking protections. As a result, we cannot determine whether observed changes reflect returning users adapting their behaviour or shifts in the underlying composition of the visitor base.

Finally, the use of PCA for dimensionality reduction, while beneficial for noise suppression and comparability across time slices, implies that clustering is performed in a reduced component space rather than directly on the original features. This improves the scalability and stability, but it reduces the direct interpretability of cluster distances and may down-weight rare behavioural patterns. Building on this reduced space, the choice of Mini-Batch K-Means as the primary clustering algorithm also imposes strong assumptions, such as the expectation that clusters are convex and linearly separable. In practice, behavioural data may contain non-linear or irregular structures that are not well captured by this geometry. Additionally, the number of clusters ($k = 6$) was fixed for each time slice based on prior work, which may have further constrained the detection of emerging or disappearing patterns.

6.2 Future Work

Future research could address these limitations in several ways. Expanding the analysis to multiple years and additional countries would enable the study of longer-term and cross-market behavioural trends. Incorporating persistent user identifiers would allow for tracking and the analysis of individual behavioural evolution.

From a methodological perspective, experimenting with alternative clustering algorithms, such as density-based methods, Gaussian Mixture Models, or spectral clustering, could uncover non-linear structures and capture more complex cluster geometries. Moreover, adaptive approaches that allow the number of clusters to vary over time might reveal emerging or disappearing behavioural types. Finally, extending the feature set with contextual variables such as device type, referral source, or campaign attribution could provide deeper insights into the drivers of behavioural change.

Overview of Generative AI Tools Used

We used OpenAI ChatGPT only as a language aid to improve grammar, wording, and clarity of text that we wrote. All suggestions were reviewed and edited. We remain responsible for all content.

List of Figures

List of Tables

Bibliography

[1] Olatz Arbelaitz, Ibai Gurrutxaga, Javier Muguerza, Jesús M. Pérez, and Iñigo Perona. An extensive comparative study of cluster validity indices. *Pattern Recognit.*, 46(1):243–256, 2013.

[2] Preeti Arora, Deepali, and Shipra Varshney. Analysis of k-means and k-medoids algorithm for big data. *Procedia Computer Science*, 78:507–512, 2016. 1st International Conference on Information Security Privacy 2015.

[3] David Arthur and Sergei Vassilvitskii. k-means++: the advantages of careful seeding. In Nikhil Bansal, Kirk Pruhs, and Clifford Stein, editors, *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2007, New Orleans, Louisiana, USA, January 7-9, 2007*, pages 1027–1035. SIAM, 2007.

[4] Dorna Bandari, Shuo Xiang, Jolie Martin, and Jure Leskovec. Categorizing user sessions at pinterest. In *IEEE International Conference on Big Data and Smart Computing, BigComp 2019, Kyoto, Japan, February 27 - March 2, 2019*, pages 1–8. IEEE, 2019.

[5] Jean-Patrick Baudry and Gilles Celeux. EM for mixtures - initialization requires special care. *Stat. Comput.*, 25(4):713–726, 2015.

[6] Christopher M. Bishop. *Pattern recognition and machine learning, 5th Edition.* Information science and statistics. Springer, 2007.

[7] H. Onur Bodur, Noreen M. Klein, and Neeraj Arora. Online price search: Impact of price comparison sites on offline price evaluations. *Journal of Retailing*, 91(1):125–139, 2015.

[8] Tadeusz Caliński and Harabasz JA. A dendrite method for cluster analysis. *Communications in Statistics - Theory and Methods*, 3:1–27, 01 1974.

[9] M. Emre Celebi, Hassan A. Kingravi, and Patricio A. Vela. A comparative study of efficient initialization methods for the k-means clustering algorithm. *Expert Syst. Appl.*, 40(1):200–210, 2013.

[10] Tsan-Ming Choi, Hing Kai Chan, and Xiaohang Yue. Recent development in big data analytics for business operations and risk management. *IEEE Trans. Cybern.*, 47(1):81–92, 2017.

[11] Simon Crase and Suresh Thennadil. An analysis framework for clustering algorithm selection with applications to spectroscopy. *PLOS ONE*, 17:e0266369, 03 2022.

[12] Ling Ding, Chao Li, Di Jin, and Shifei Ding. Survey of spectral clustering based on graph theory. *Pattern Recognit.*, 151:110366, 2024.

[13] Gregory Dobler, Jordan Vani, and Trang Tran Linh Dam. Patterns of urban foot traffic dynamics. *Comput. Environ. Urban Syst.*, 89:101674, 2021.

[14] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In Evangelos Simoudis, Jiawei Han, and Usama M. Fayyad, editors, *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), Portland, Oregon, USA*, pages 226–231. AAAI Press, 1996.

[15] Javier Fabra, Pedro Álvarez, and Joaquín Ezpeleta. Log-based session profiling and online behavioral prediction in e-commerce websites. *IEEE Access*, 8:171834–171850, 2020.

[16] Stephen Few. *Information dashbord design - the effective visual communication of data.* O'Reilly, 2006.

[17] Alexander Gepperth and Benedikt Pfülb. Gradient-based training of gaussian mixture models for high-dimensional streaming data. *Neural Process. Lett.*, 53(6):4331–4348, 2021.

[18] Miguel Alves Gomes and Tobias Meisen. A review on customer segmentation methods for personalized customer targeting in e-commerce use cases. *Inf. Syst. E Bus. Manag.*, 21(3):527–570, 2023.

[19] Gregory James Hamerly. *Learning Structure and Concepts in Data Through Data Clustering.* PhD thesis, University of California, San Diego, 2003.

[20] Mariya Hendriksen, Ernst Kuiper, Pim Nauts, Sebastian Schelter, and Maarten de Rijke. Analyzing and predicting purchase intent in e-commerce: Anonymous vs. identified customers. *CoRR*, abs/2012.08777, 2020.

[21] Harold Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24:417–441, 1933.

[22] Anil K. Jain. Data clustering: 50 years beyond k-means. *Pattern Recognit. Lett.*, 31(8):651–666, 2010.

[23] Anil K. Jain, M. Narasimha Murty, and Patrick J. Flynn. Data clustering: A review. *ACM Comput. Surv.*, 31(3):264–323, 1999.

[24] Chris Janiszewski. The influence of display characteristics on visual exploratory search behavior. *Journal of Consumer Research*, 25(3):290–301, 1998.

[25] Ian Jolliffe and Jorge Cadima. Principal component analysis: A review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374:20150202, 04 2016.

[26] Kwon Jung, Yoon C. Cho, and Sun Lee. Online shoppers' response to price comparison sites. *Journal of Business Research*, 67(10):2079–2087, 2014.

[27] Kamran Khan, Saif ur Rehman, Kamran Aziz, Simon Fong, Sababady Sarasvady, and Amrita Vishwa. DBSCAN: past, present and future. In *The Fifth International Conference on the Applications of Digital Information and Web Technologies, ICADIWT 2014, Chennai, India, February 17-19, 2014*, pages 232–238. IEEE, 2014.

[28] Harold W. Kuhn. The hungarian method for the assignment problem. In Michael Jünger, Thomas M. Liebling, Denis Naddef, George L. Nemhauser, William R. Pulleyblank, Gerhard Reinelt, Giovanni Rinaldi, and Laurence A. Wolsey, editors, *50 Years of Integer Programming 1958-2008 - From the Early Years to the State-of-the-Art*, pages 29–47. Springer, 2010.

[29] Juhnyoung Lee, Mark Podlaseck, Edith Schonberg, and Robert Hoch. Visualization and analysis of clickstream data of online stores for understanding web merchandising. *Data Min. Knowl. Discov.*, 5(1/2):59–84, 2001.

[30] Pawan Lingras, Mofreh Hogo, and Miroslav Snorek. Temporal cluster migration matrices for web usage mining. In *2004 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2004), 20-24 September 2004, Beijing, China*, pages 441–444. IEEE Computer Society, 2004.

[31] Pawan Lingras, Rui Yan, and Mofreh Hogo. Rough set based clustering: Evolutionary, neural, and statistical approaches. In Bhanu Prasad, editor, *Proceedings of the 1st Indian International Conference on Artificial Intelligence, IICAI 2003, Hyderabad, India, December 18-20, 2003*, pages 1074–1087. IICAI, 2003.

[32] Stuart P. Lloyd. Least squares quantization in PCM. *IEEE Trans. Inf. Theory*, 28(2):129–136, 1982.

[33] Gaurav Misra, Matteo Migliavacca, and Fernando E. B. Otero. Behavioural user identification from clickstream data for business improvement. In Max Bramer and Richard Ellis, editors, *Artificial Intelligence XXXVIII - 41st SGAI International Conference on Artificial Intelligence, AI 2021, Cambridge, UK, December 14-16, 2021, Proceedings*, volume 13101 of *Lecture Notes in Computer Science*, pages 341–354. Springer, 2021.

[34] Wendy W. Moe. Buying, searching, or browsing: Differentiating between online shoppers using in-store navigational clickstream. *Journal of Consumer Psychology*, 13(1):29–39, 2003. Consumers in Cyberspace.

[35] Wendy W. Moe and Peter S. Fader. Capturing evolving visit behavior in clickstream data. *Journal of Interactive Marketing*, 18(1):5–19, 2004.

[36] Amrutha Muralidhar and Yathindra Lakkanna. From clicks to conversions: Analysis of traffic sources in e-commerce. *CoRR*, abs/2403.16115, 2024.

[37] Kevin P. Murphy. *Machine learning - a probabilistic perspective*. Adaptive computation and machine learning series. MIT Press, 2012.

[38] Rainer Olbrich and Christian Holsing. Modeling consumer purchasing behavior in social shopping communities with clickstream data. *Int. J. Electron. Commer.*, 16(2):15–40, 2011.

[39] Mahamed G. H. Omran, Andries P. Engelbrecht, and Ayed A. Salman. An overview of clustering methods. *Intell. Data Anal.*, 11(6):583–605, 2007.

[40] Gautam Pal, Katie Atkinson, and Gangmin Li. Real-time user clickstream behavior analysis based on apache storm streaming. *Electron. Commer. Res.*, 23(3):1829–1859, 2023.

[41] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake VanderPlas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. Scikit-learn: Machine learning in python. *J. Mach. Learn. Res.*, 12:2825–2830, 2011.

[42] Orit Raphaeli, Anat Goldstein, and Lior Fink. Analyzing online consumer behavior in mobile and PC devices: A novel web usage mining approach. *Electron. Commer. Res. Appl.*, 26:1–12, 2017.

[43] Peter J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[44] Daniel Schellong, Jan Kemper, and Malte Brettel. Clickstream data as a source to uncover con-sumer shopping types in a large-scale online setting. In *24th European Conference on Information Systems, ECIS 2016, Istanbul, Turkey, June 12-15, 2016*, page Research Paper 1, 2016.

[45] Erich Schubert, Jörg Sander, Martin Ester, Hans-Peter Kriegel, and Xiaowei Xu. DBSCAN revisited, revisited: Why and how you should (still) use DBSCAN. *ACM Trans. Database Syst.*, 42(3):19:1–19:21, 2017.

[46] Gregory Schwartzman. Mini-batch k-means terminates within $o(d/\varepsilon)$ iterations. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023.

[47] D. Sculley. Web-scale k-means clustering. In Michael Rappa, Paul Jones, Juliana Freire, and Soumen Chakrabarti, editors, *Proceedings of the 19th International Conference on World Wide Web, WWW 2010, Raleigh, North Carolina, USA, April 26-30, 2010*, pages 1177–1178. ACM, 2010.

[48] Qiang Su and Lu Chen. A method for discovering clusters of e-commerce interest patterns using click-stream data. *Electron. Commer. Res. Appl.*, 14(1):1–13, 2015.

[49] Cheng Tang and Claire Monteleoni. Convergence rate of stochastic k-means. In Aarti Singh and Xiaojin (Jerry) Zhu, editors, *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*, volume 54 of *Proceedings of Machine Learning Research*, pages 1495–1503. PMLR, 2017.

[50] Luca Vassio, Idilio Drago, Marco Mellia, Zied Ben-Houidi, and Mohamed Lamine Lamali. You, the web, and your device: Longitudinal characterization of browsing habits. *ACM Trans. Web*, 12(4):24:1–24:30, 2018.

[51] Kevin Voges, Nigel Pope, and Mark Brown. Cluster analysis of marketing data examining online shopping orientation: A comparison of k-means and rough clustering approaches. 01 2002.

[52] Ulrike von Luxburg. A tutorial on spectral clustering. *Stat. Comput.*, 17(4):395–416, 2007.

[53] Ahmadou Wagne and Julia Neidhardt. What to compare? towards understanding user sessions on price comparison platforms. In Tommaso Di Noia, Pasquale Lops, Thorsten Joachims, Katrien Verbert, Pablo Castells, Zhenhua Dong, and Ben London, editors, *Proceedings of the 18th ACM Conference on Recommender Systems, RecSys 2024, Bari, Italy, October 14-18, 2024*, pages 1158–1162. ACM, 2024.

[54] Gang Wang, Xinyi Zhang, Shiliang Tang, Haitao Zheng, and Ben Y. Zhao. Unsupervised clickstream clustering for user behavior analysis. In Jofish Kaye, Allison Druin, Cliff Lampe, Dan Morris, and Juan Pablo Hourcade, editors, *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, San Jose, CA, USA, May 7-12, 2016*, pages 225–236. ACM, 2016.

[55] Jishang Wei, Zeqian Shen, Neel Sundaresan, and Kwan-Liu Ma. Visual cluster exploration of web clickstream data. In *7th IEEE Conference on Visual Analytics Science and Technology, IEEE VAST 2012, Seattle, WA, USA, October 14-19, 2012*, pages 3–12. IEEE Computer Society, 2012.

[56] Toshihiko Yamakami. Markov model based mobile clickstream analysis with sub-day, day and week-scale transitions. In Bruno Apolloni, Robert J. Howlett, and Lakhmi C. Jain, editors, *Knowledge-Based Intelligent Information and Engineering Systems, 11th International Conference, KES 2007, XVII Italian Workshop on Neural Networks, Vietri sul Mare, Italy, September 12-14, 2007, Proceedings, Part III*, volume 4694 of *Lecture Notes in Computer Science*, pages 507–513. Springer, 2007.

[57] Melina Zavali, Ewelina Lacka, and Johannes De Smedt. Shopping hard or hardly shopping: Revealing consumer segments using clickstream data. *IEEE Trans. Engineering Management*, 70(4):1353–1364, 2023.

[58] Fuguo Zhang, Shumei Qi, Qihua Liu, Mingsong Mao, and An Zeng. Alleviating the data sparsity problem of recommender systems by clustering nodes in bipartite networks. *Expert Syst. Appl.*, 149:113346, 2020.