

**Projektbericht zum Modul Information Retrieval und
Visualisierung Sommersemester 2022**

Marktanalyse des Videospielsemarktes

**Analyse und Visualisierung der Verkaufszahlen der Videospiele auf der Plattform
XBoxOne**

Lena Arloth

Matrikelnummer: 217207784

9. Dezember 2022

Link zum GitHub-Repository:

<https://github.com/Lena-Ar/Info-Vis>

Letzter Commit:

()

Link zur öffentlich zugänglichen Website (GitHub-Pages):

<https://lena-ar.github.io/Info-Vis/>

Inhaltsverzeichnis

1	Einleitung	1
1.1	Anwendungshintergrund	1
1.2	Zielgruppe	2
1.3	Überblick und Beiträge	3
2	Daten	3
2.1	Bereitstellung und Vorverarbeitung der Daten	4
3	Visualisierungen	5
3.1	Analyse der Anwendungsaufgaben	5
3.2	Anforderungen an die Visualisierungen	7
3.3	Präsentation der Visualisierungen	8
3.3.1	Visualisierung Eins	8
3.3.2	Visualisierung Zwei	9
3.3.3	Visualisierung Drei	10
3.4	Interaktion	12
4	Implementierung	13
4.1	Data.elm	13
4.2	TreeHierarchy.elm	14
4.3	ParallelPlot.elm	15
4.4	Scatterplot.elm	16
4.5	MainScatterParallel.elm	17
5	Anwendungsfälle	19
5.1	Anwendung Visualisierung Eins	19
5.2	Anwendung Visualisierung Zwei	20
5.3	Anwendung Visualisierung Drei	21
6	Verwandte Arbeiten	22
7	Zusammenfassung und Ausblick	24

1 Einleitung

Der weltweite Videospielmarkt über alle Plattformen hinweg ist mit einem Volumen von 156,17 Milliarden Euro im Jahre 2021 sehr groß und wird 2022 voraussichtlich auf 176,13 Milliarden Euro steigen.[11] Entsprechend viel Geld und Potenzial durch Wachstum verbirgt sich in ihm. Gleichzeitig benötigt es für die Entwicklung neuer Videospiele hohe Investitionen vonseiten der Entwicklerstudios und Verleger, folgend Publisher genannt, die erst nach einer gewissen Sicherheit durch Analyse des Marktes getroffen werden sollten. Um Prognosen und Tendenzen für die Zukunft finden zu können, braucht es die Erkenntnisse aus vergangenen Daten. Eine Möglichkeit hierzu ist die Analyse der totalen Verkaufszahlen der Videospieltitel aufgeschlüsselt nach Genre weltweit.

Als Problemstellung ergibt sich das fehlende Wissen der Entscheider innerhalb der Publisher, in welche Videospieltitel in welchem Genre und mit welcher Regionsausrichtung sie künftig investieren sollten, um das Potenzial des großen und wachsenden Videospielmarktes möglichst gut zu nutzen. Investition bezieht sich hierbei nicht zwingend auf eine konkrete Entscheidung für die Auftragsvergabe eines neuen Titels, sondern vor allem auf die Veranlassung detaillierter, umfassender Marktstudien. Erst dann kann eine fundierte endgültige Entscheidung getroffen werden. Es ergibt sich eine durchzuführende grobe Marktanalyse der Verkaufszahlen einer Plattform in der Videospielindustrie aus Sicht der Publisher. Sie fungiert als Grundlage und Unterstützung weiterer Entscheidungen und Investitionen.

Die Relevanz ergibt sich aus zuvor Beschriebenem und der Notwendigkeit von Marktanalysen als Voraussetzung für unternehmerische Entscheidungen, Chancen und Risiken sowie Potenzialen.[5] Es ist für Unternehmen wie Publisher sehr wichtig zu erfahren, was und in welchem Genre künftig mit guten Verkaufschancen entwickelt werden soll. Weiterhin ist es relevant, mittels einer Marktanalyse zu erfahren, wie das Unternehmen selbst im Markt aufgestellt ist und was die Konkurrenz bietet. Um dies zu erreichen, ist es wichtig, Muster, Cluster und Korrelationen zwischen den Verkaufszahlen verschiedener Regionen in einem Genre zu erkennen. Für ein erleichtertes Verständnis sowie das Hervorheben von Auffälligkeiten sind Techniken aus dem Bereich der Visual Analytics von kritischer Relevanz.

Es leitet sich folgende Fragestellung ab, die es mittels einer groben Marktanalyse aufbereitet, verdeutlicht und sichtbar gemacht durch Visualisierungstechniken zu beantworten gilt: In welches Genre und mit welchem Fokus auf die Regionen der Welt sollte ein Publisher künftig mit der Beauftragung detaillierter, teurerer Marktstudien zur Entscheidung über die Auftragsvergabe neuer Videospieltitel investieren?

1.1 Anwendungshintergrund

Marktanalysen sind für Unternehmen wie Publisher von zentralem Wert, da sie mit ihnen strategische Zukunftsfragen zur Ausrichtung beantworten können. Sie müssen unter anderem Fragen zu Produktverbesserungen, -erweiterungen oder -neuerungen, Investitionen in diese und Poten-

ziale und Risiken durch Korrelationen, Muster und Ausreißer beantworten können.[5] Dabei kann sie detaillierter oder oberflächlicher sein. In dieser Anwendung ist eine oberflächliche Analyse zur Ersparnis von Zeit und Geld geplant, mittels derer monetär interessante Genres und Regionen identifiziert werden können. Aufgründessen kann eine Investition in eine detaillierte, deutlich aufwendigere und damit teure Marktstudie in den zuvor identifizierten Genres und mitunter Regionen in Auftrag gegeben werden. Diese kann zu einer deutlich fundierteren Bestätigung oder Ablehnung der Vorabentscheidung beitragen.

Der Videospiegelmarkt ist in verschiedene Segmente geteilt, die wiederum unterteilt sind. Für diese Arbeit wird das Segment der Spieleplattform und Konsole *XBoxOne* von Microsoft gewählt. Dies liegt zum einen in der Verfügbarkeit der Daten begründet und zum anderen in der Beliebtheit der Konsole und der Konkurrenz zu anderen Plattformen wie *Playstation 4* und *Computer*. Gleichzeitig ergeben Analysen innerhalb einer Plattform für die Entscheidungsunterstützung künftiger Investitionen in Titel dieser Plattform mehr Sinn, da die Plattformen untereinander nicht kompatibel sind.

Die Konsole *XBoxOne* von Microsoft verkaufte sich besonders in den Jahren 2013 bis Anfang 2019 gut mit einer Spitze im Dezember 2015 mit rund 715.000 verkauften Einheiten im Monat. Mittlerweile ist die Beliebtheit der Konsole zwar stark zurückgegangen, doch lassen sich aus den Verkaufszahlen von Videospielen ihrer Plattform weiterhin Erkenntnisse für die Zukunft und potenzielle Investitionen der Publisher ableiten.[16] Hierbei geht es auch um Investitionen in neue Titel für die *XBoxOne*, aber vor allem um solche in die Nachfolgekonsolen *XBox Series X* und *XBox Series S*. Durch die Abwärtskompatibilität scheint es wahrscheinlich, dass Nutzer der *XBoxOne* auch die Nachfolgekonsolen nutzen.[12] Somit behalten die Verkaufszahlen der *XBoxOne* ihre wenn auch leicht verminderte Aussagekraft. Der Sinn einer übersichtsartigen, grobgranularen Marktanalyse dieses Segments bleibt bestehen. Zur besseren Vergleichbarkeit und Aussagekraft der Verkaufszahlen und ihrer Auffälligkeiten der Videospiele ist eine Filterung nach Genre sinnvoll.

1.2 Zielgruppe

Die in dieser Arbeit betrachtete Zielgruppe setzt sich aus Entscheidern der Publisher von Videospielen zusammen. Diese Entscheider sind vorrangig Personen des oberen Management, aber auch des mittleren Managements, vorrangig aus dem Bereich Marketing und Forschung und Entwicklung. Das mittlere Management übernimmt dabei neben dem Mitentscheiden auch die Aufgabe des Präsentierens der Analysen mittels Visualisierungen für das obere Management. Weiterhin zählen in begrenztem Rahmen auch Stakeholder der Publisher zur Zielgruppe, da sie über Entscheidungen des Unternehmens zu künftigen Strategien informiert werden sollten.

In der Zielgruppe existiert kein detailliertes bzw. stark spezifiziertes Vorwissen zu speziellen Visualisierungstechniken. Jedoch ist anzunehmen, dass sich vor allem das mittlere und obere Management mit in der Betriebswirtschaftslehre häufiger vorkommenden Visualisierungen auskennt und diese ohne viele Erklärungen auswerten kann. Dazu zählen neben Balken- und

Kreisdiagramme, Box Plots und Zeitreihendiagramme auch Scatterplots, explizite Baumdiagramme und Parallele Koordinaten Plots. Bei spezielleren Visualisierungstechniken wären mehr Erklärungen sowie mehr Zeit zur Analyse und Entscheidung nötig. Dies wäre für eine möglichst schnelle und kostengünstige Entscheidung zu detaillierteren Marktstudien, wobei noch keine konkrete Entscheidung für einen Titel getroffen wird, nicht zielführend.

Durch die Visualisierungen werden mehrere Informationsbedürfnisse adressiert. Erstens wird das Bedürfnis nach übersichtsartigen Informationen zur Videospielindustrie im Sinne der eigenen Position und der Konkurrenz für die *XBoxOne* bezüglich der Verbindungen von Publisher, Genre und Videospiel adressiert. Zweitens werden Informationsbedürfnisse nach Zusammenhängen und Mustern zwischen Regionen in den jeweiligen Genres befriedigt, um daraus strategische Entscheidungen ableiten zu können. Drittens wird das Bedürfnis nach konkreten Verkaufszahlen der einzelnen Spieletitel angesprochen.

1.3 Überblick und Beiträge

Die in dieser Arbeit verwendeten Daten stammen aus einem Datensatz für die *XBoxOne* mit den Attributen Tabellenposition, Publisher, Jahr der Veröffentlichung, Genre, Verkaufszahlen global sowie in den Regionen Nordamerika, Europa, Japan und Rest der Welt von der Plattform Kaggle.[15] Die Visualisierungen entwickeln sich vom Groben zum Detaillierten. In einer ersten Visualisierung wird zur Unterstützung einer Übersicht über die Publisher, Genre und Videospieltitel ein hierarchisches, explizites Baumdiagramm verwendet. Dadurch wird zudem ein Ansatzpunkt zur Auswahl der für die Publisher zu untersuchenden Genres gegeben. Zur Erkennung von Mustern und Ausreißern in den Verkaufszahlen eines Genres auch über alle Regionen hinweg bzw. in einer Ansicht aller Dimensionen gleichzeitig dient der folgende Parallele Koordinaten Plot. Zuletzt können in der dritten Visualisierung, dem Scatterplot, Korrelationen zwischen zwei Regionen sowie Ausreißer genauer betrachtet werden. In allen Visualisierungen sind Spieletitel mit Detailinformationen zu Publishern und Verkaufszahlen für einen konkreten Vergleich erkennbar. Mittels aller Visualisierungen kann die übergeordnete Fragestellung beantwortet werden, welches Genre durch positive Korrelationen, Tendenzen zu wenig Konkurrenz oder hohen Spielerzahlen durch viele Titel in dem Genre und der Analyse von Mustern und Ausreißern Potenzial bietet und damit detaillierterer Marktanalysen bedarf.

2 Daten

Der genutzte Datensatz "*Video Games Sales Dataset*" entstammt der Plattform Kaggle.[15] Die Daten entstanden laut SID_TWR durch die von Gregory Smith motivierte Erweiterung eines Web Scrapes von *VGChartz Video Games Sales* um Attribute eines Web Scrapes von *Metacritic*. Aus den drei zur Verfügung gestellten Datensätzen wird *XBoxOne_GameSales* für dieses Projekt gewählt. Die erwähnten Erweiterungen um Attribute von *Metacritic* sind hier nicht enthalten. Der Originaldatensatz liegt als CSV-Datei vor und beinhaltet zehn Spalten mit 613 einzelnen

Positionen.

Zu jeder *Position* sind der Videospielname sowie das jeweilige Jahr der Veröffentlichung aufgelistet. *Genre* kategorisiert die Videospiele entsprechend und *Publisher* ordnet jedem Videospiel seinen Verleger zu. Die Attribute *North America*, *Europe*, *Japan*, *Rest of World* und *Global* stellen die Verkäufe der Videospiele in *millions of units*, also Millionen Stück verkaufter Kopien, in diesen Regionen dar. Der gewählte Datensatz bildet Videospiele von 2013 bis circa 2020 ab. Aufgrund fehlender Aktualisierung der Verkaufszahlen auf *VGChartz Video Games Sales* ab 2018 ist davon auszugehen, dass die für das Projekt verwendeten Daten kumulierte Verkaufszahlen ab 2013 bis circa 2018 abbilden.

Die vorhandenen Daten werden als gut geeignet für die Zielgruppe und das zuvor eingeleitete Zielproblem eingeschätzt. Sie ermöglichen eine grobe Übersicht über die verschiedenen Publisher, ihre Videospiele und bedienten Genres. Weiterhin schlüsseln sie detailliert auf, wie sich ein Videospiel kumuliert seit seiner Veröffentlichung global, aber auch in den einzelnen Weltregionen verkaufte.

2.1 Bereitstellung und Vorverarbeitung der Daten

Die originale CSV-Datei wird zur Bearbeitung in *Open Office Calc* in das ODS-Format überführt und mit allen modifizierten Datendateien, dem Quellcode und dem Bericht in einem öffentlichen GitHub Repository bereitgestellt. Die Datendateien sind im Ordner *Daten* in den Unterordnern *CSV*, *JSON* und *Tabelle* je nach Format auffindbar.

Es wird eine JSON-Datei benötigt, in der die Beziehungen der Publisher, Genre und Videospiele zueinander beschrieben sind. Videospiele seien Kinder der Genres, die wiederum Kinder der Publisher sind und unter einem Wurzelknoten zusammengefasst werden. Zusätzlich wird für weitere Visualisierungen eine CSV-Datei mit nachfolgend beschriebenen Modifikationen benötigt.

Der Datensatz enthält fehlende Werte bei Publishern und Verkaufszahlen jeder Region sowie für die Anwendung nicht benötigte Informationen. Zur besseren Nachvollziehbarkeit und Dokumentation des Ablaufs der Datenvorverarbeitung befindet sich im GitHub Repository im Ordner *Daten* eine zweite README.md-Datei. Die benötigte Datengrundlage wird in sieben Schritten erreicht, die folgend grob beschrieben sind und detailliert in angesprochener README nachvollzogen werden können.

Aus der Originaldatei und ihrer konvertiertes ODS-Pendant werden Testversionen der Originaldaten mit den ersten 20 Positionen im CSV- und ODS-Format erstellt. Sie werden um die Spalte des Erscheinungsjahres reduziert und dienen der Übersichtlichkeit und Funktionalität in der Entwicklungsphase.

Es folgt die eigentliche Modifikation der Originaldaten unter Erstellung neuer Dateien zur Bewahrung der Transparenz. Das Erscheinungsjahr des jeweiligen Videospiels wird aus der durchzuführenden Marktanalyse und ihrer Visualisierung ausgegliedert und gelöscht. Es können weder Daten für ein Zeitreihendiagramm extrahiert werden, noch ergibt ein Vergleich der kumulierten

Verkaufszahlen in Relation zum Erscheinungsjahr für die Problemstellung und deren Lösung keinen Sinn. Weiterhin werden jene Positionen eliminiert, die einen Wert von Null in den Verkaufszahlen aller Regionen bzw. automatisch global aufweisen. Sie sind fehlerhaft und haben in der Visualisierung keine Verwendung. Zum Ausschluss von Fehlern bei der manuellen Vorverarbeitung sowie universelleren Einsetzbarkeit des Quellcodes und Änderung der Rohdaten wird dieser Schritt zusätzlich in Elm programmiert. Aufgrund von Irrelevanz werden jene Positionen gelöscht, denen kein Publisher zugeordnet ist oder dessen Wert *Unknown* ist. Auch Positionen stammend von einem Publisher nur dieses Videospiel werden ausgegliedert. Es ist von einer sehr spezifischen Ausrichtung jener Publisher auszugehen, die keine hier angestrebte übersichtsartige Marktanalyse benötigen. Weiterhin wird durch Umbenennung 2014 der Name des Publishers *Namco Bandai Games* in *Bandai Namco Games* vereinheitlicht.

Die Konvertierung der für die Beziehungen erstellten CSV-Datei in eine JSON-Datei erfolgt mittels des Tools *convertcsv.com*. Diese wird auf Fehler kontrolliert, der Wurzelknoten hinzugefügt sowie die entstandenen leeren Felder gelöscht. Zur Erzielung der gewünschten Visualisierung der hierarchischen Struktur wird pro Genre der jeweilige Eltern-Publisher wenn möglich sinnvoll abgekürzt in Klammern hinzugefügt. Zusätzlich wird eine Testdatei mit 20 Videospielen erstellt.

Eine Umrechnung der Einheiten der Verkäufe ist durch schon gegebene Übersichtlichkeit nicht ratsam. Zur Erläuterung dienen Informationstexte und die passenden Achsenbeschriftungen.

Die Filterung der Positionen, die in nur einer Region Null-Werte aufzeigen, wurde in Betracht gezogen, jedoch verworfen. Diese Positionen würden nicht angezeigt, obwohl auch Informationen über keine erfolgten Verkäufe und die entsprechende Region relevant sind. Es ist zu beachten, dass als Null-Werte auch solche aufgeführt sind, deren Verkaufszahlen bei unter 0.01 Millionen Stück verkaufter Videospieldkopen liegen.

3 Visualisierungen

Nachfolgend wird eine Analyse der Anwendungsaufgaben zur Lösung des Zielproblems durchgeführt, Anforderungen an die Visualisierungen abgeleitet und diese präsentiert.

3.1 Analyse der Anwendungsaufgaben

Das Zielproblem der konkreten Anwendungsaufgaben ist eine grobgranulare Marktanalyse des Videospieldmarktes der *XBoxOne* als Entscheidungsunterstützung zur Investition in detaillierte Studien und folglich neue Videospiele. Fokussiert wird die Frage, welche Genres und möglicherweise Spezifizierungen auf bestimmte Regionen dort untersucht werden sollen. Zur Problemlösung sollen sowohl die Publisher und ihre Konkurrenz betrachtet werden als auch Verkaufschancen, Zusammenhänge und Auffälligkeiten in den einzelnen Genres zwischen den Regionen der Welt. Mit Rücksicht auf die Zielgruppe ist eine sprachlich von jedem zu verstehende Lösung im Kontext von Präsentationen sowie Kosten- und Zeitminimierung wichtig. Das intuitive, schnelle und eindeutig bei allen Managern der Publisher Entstehen der mentalen Modelle zur ein-

heitlichen Analyse der Daten gerade bei Präsentationen oder dem kurzfristigen Gebrauch der Visualisierungen ist essenziell, da schnelle Entscheidungen Wettbewerbsvorteile ermöglichen.

Als erste Anwendungsaufgabe ergibt sich die Schaffung eines Marktüberblicks als Ansatzpunkt weiterer Entscheidungen und erste Eingrenzungen der zu untersuchenden Genres. Das Management der Publisher benötigt eine Übersicht über eigene und konkurrierende Angebote, die insbesondere für größere, breit aufgestellte Verleger sinnvoll ist. Es sollten die Publisher, ihre abgedeckten Genres und die in ihnen kategorisierten Videospieltitel ersichtlich sein. Je nach Strategie des betrachtenden Publishers können bspw. aufgrund der geringen Anzahl eigener oder hoher Anzahl konkurrierender Spiele in einem Genre Vorentscheidungen zur Eingrenzung des mittels nachfolgender Anwendungsaufgabe näher zu untersuchenden Genres getroffen werden. Zur Lösung der Anwendungsaufgabe soll ein mentales Modell von bildlich dargestellten Beziehungen und Hierarchien zwischen Publishern, Genre und Titel entstehen. Dies könnte durch schriftliche Repräsentationen umgangen werden, jedoch unterstützt das mentale Modell das Verständnis und beschleunigt den Vorentscheid ungemein. Besonders im Rahmen einer Präsentation ist es unumgänglich.

Als zweite Anwendungsaufgabe ergibt sich die Entscheidung für eine Investition in ein Genre mit Bezug auf die Regionen mittels einer Analyse von Verkaufschancen, Zusammenhänge und Auffälligkeiten in diesem. Es ist wichtig, erste Auffälligkeiten in allen Regionen der Welt zu erkennen, um ein Genre für eine Investition in eine folgende Marktstudie zu fokussieren. Dazu zählen Spiele, die auffällig gut verkauft wurden und eine starke Konkurrenz darstellen sowie Muster in den Verkaufszahlen, die aus mehreren Videospielen im Genre in den Regionen der Welt gebildet werden wie bspw. viele Spiele, die in allen Regionen außer einer mittelmäßig gut verkauft wurden. Weiterhin ist es für eine mögliche Spezifikation der künftigen Marktstudie auf eine Region sinnvoll, diese Muster genauer durch Untersuchung nur besonders interessanter Regionen zu prüfen. Somit können Abhängigkeiten besonders zwischen zwei Regionen detailliert analysiert werden. Benötigt wird zudem eine Erkennbarkeit der konkreten Informationen der Spieletitel. Bei der Bearbeitung können zwei mentale Modelle helfen. Zum einen soll eine Art Sicht von oben auf die Welt mit allen Regionen entstehen, die durch die Videospiele als Datenpunkte verbunden sind. Durch Häufungen von Videospielen und Ähnlichkeiten in den Verbindungen vieler Spiele sollen mental Muster und Cluster über alle Regionen entstehen. Zum anderen ist ein mentales Modell mit erhöhtem Detaillierungsgrad zur Lösung der Anwendungsaufgabe sinnvoll, mit dem je zwei zuvor als interessant erkannte Regionen vergrößert werden. Hierbei sollen durch Punkte im Raum die Verkaufszahlen mental positioniert und damit vergleichbar werden sowie Korrelationen in ihrer Art leichter bestimmbar sein. Visualisierungen zum Aufbau dieser mentalen Modelle sind elementar. Ohne sie sind Analysen der gewünschten Aspekte teils nur sehr schwer möglich sind.

Die Analyse der Anwendungsaufgaben ergibt möglichst geringe monetäre und zeitliche Aufwände zur Erstellung der Visualisierungen zur Unterstützung der mentalen Modelle. Dadurch kann die gewünschte Kostenersparnis durch eine vorab günstigere, grobgranulare Marktanaly-

se zur Investitionsentscheidung in detaillierte, teure und zeitintensive Marktanalysen höheren Umfangs erreicht werden.

3.2 Anforderungen an die Visualisierungen

Über alle Anwendungsaufgaben hinweg stellen sich die Anforderungen nach Expressivität, Effektivität und Angemessenheit nach Schuman und Müller.[14, 9ff.] Für die gewünschte Expressivität ist das Verständnis der Visualisierungen ohne Missverständnisse oder Doppel- bzw. Mehrdeutigkeiten der Daten nötig, sodass nur die in den Daten enthaltenen Aussagen visualisiert werden. Es werden drei visuelle Ebenen der Informationen definiert, von denen hier die obere Ebene zutrifft.[1] Für eine gute Expressivität im Rahmen dieser ist die Visualisierung expliziter Fakten zu Videospielen und Verkaufszahlen sowie versteckte Zusammenhänge nötig. Wie zuvor analysiert sind hier vor allem die Offenlegung von Hierarchien und Mustern, Abhängigkeiten und Ausreißern zwischen Regionen in einem Genre zu nennen. Zur Erfüllung der Effektivität ist ein intuitives, schnelles Verständnis der Visualisierungen gefordert. Die Stellenwert dieser Anforderung begründet sich im zuvor analysierten Zeitdruck durch die potenzielle Anwendung in Präsentationen sowie der Entscheidungen. Aufgrund der mit wenig spezifischem Vorwissen in den Visual Analytics ausgestatteten Zielgruppe bedeutet Effektivität auch eine geringere Komplexität der Darstellungen, ergo intuitiv für die Zielgruppe verständliche Visualisierungen vor allem von Korrelationen und Mustern. Die der Zielgruppe zugrunde liegende minimal verfügbare Zeit zum Verständnis legt den Fokus klar auf zeitlicher Effizienz bei bestmöglicher Effektivität. Durch die angestrebte Nutzung der Visualisierungen zur Entscheidungsunterstützung des Fokus weiterer Marktstudien ist besonders die Anforderung an Angemessenheit zentral. Die Visualisierungen müssen für einen Vorteil im vorgestellten Verfahren wenig zeit- und kostenaufwendig sein. Dazu ist es notwendig, den Umfang und Aufwand der Visualisierungen zum Erreichen der Expressivität für die Darstellung der oberen visuellen Ebene der Informationen auf niedrigerem Niveau zu halten.

Eine Rücksichtnahme auf mögliche Sehschwächen der Zielgruppe durch durchdachte Farbgebung von Visualisierungen und Website ist nötig. Ohne Expressivität und Effektivität negativ zu beeinflussen, ist eine Beschränkung auf eine Farbgruppe sinnvoll. Untergeordnet sind außerdem Anforderungen an Ästhetik und indirekte Leitung des Blicks über die Website durch gerade erkennbare Abstufungen in Farbe und Kontrast, sogenannte *Just Noticable Differences*, zu erfüllen. Für ein einheitliches Verständnis der Visualisierungen der global agierenden Zielgruppe wird die Sprache Englisch festgelegt.

Die Analyse der zweiten Anwendungsaufgabe ergab die Notwendigkeit der Auswahl möglicher Genres zur Entscheidung für Investitionen in Marktstudien in bestimmten Genres. Folglich gilt die Anforderung an einen Filter zur Auswahl der Genres in den zwei Visualisierungen der Verkaufszahlen. Aus der Analyse jener zweiten Anwendungsaufgabe folgt außerdem die Anforderung an die dritte Visualisierung mit der Auswahl von zwei Regionen zum Vergleich.

Konkret muss die erste Visualisierung als Ausgangspunkt der Anwendung die in der Analyse

der ersten Anwendungsaufgabe geforderten Beziehungen zwischen Publishern, Genres und Titel abbilden und auf einen Blick für die Zielgruppe sichtbar machen. An die zweite Visualisierung stellt sich die Anforderung nach einer ganzheitlichen Sicht über alle Attribute der Regionen sowie die Aufdeckung von in den Verkaufszahlen der Regionen versteckten Mustern, Clusterbildungen und möglichen Ausreißern über alle Regionen hinweg. Die Abbildung mehrdimensionaler Daten ist essenziell. Zur gewünschten Vergrößerung der vorherigen Ansicht auf zwei interessante Attribute genügt die Ansicht zweidimensionaler Daten für die dritte Visualisierung. Jedoch müssen die Attribute resp. Regionen variabel wählbar sein. An die dritte Visualisierung wird vor allem die Anforderung nach detaillierter Darstellung der Art der Korrelationen gestellt. Nützlich ist eine Überprüfung der in der zweiten Visualisierung sichtbaren Muster und Cluster in den zwei gewählten Dimensionen.

3.3 Präsentation der Visualisierungen

Es werden drei Visualisierungstechniken zur Erfüllung der Anforderungen gewählt und präsentiert. Zur Datenpunktmarkierung und -hervorhebung sowie der Leitung des Blicks über die Website werden neben Schwarz-Weiß werden zwei harmonische Grüntöne in unterschiedlichen Kontrasten und Opazitäten genutzt.

3.3.1 Visualisierung Eins

Zur Erfüllung der Anforderungen an die Visualisierung der ersten Anwendungsaufgabe wird ein explizites Baumdiagramm gewählt. Mittels dieser können hierarchische Beziehungen zwischen Attributwerten als Knoten in Form eines Baumes mit einem Wurzelknoten abgebildet werden. Hierarchische Beziehungen zwischen den Knoten seien durch gerade Linien, Kurven oder Bögen gekennzeichnet, die Knoten miteinander verbinden.[13, S. 1]

Pfade zwischen den Knoten sowie die Positionierung von Eltern über Kindknoten stellen hier die hierarchische Beziehung dar. Publisher sind Elternknoten der Genres, die wiederum Elternknoten der Videospieltitel sind. Um ungewollten Kreuzungen der Pfade zwischen Publishern und Genres zu verhindern, wird letzteren zur eindeutigen Identifikation das Kürzel des zugehörigen Publishers hinzugefügt. Andernfalls verlöre die Visualisierung ihren Zweck der Übersicht und einfachen Betrachtung der eigenen abgedeckten Genres und in ihnen verordnete Titel sowie der Vergleich mit der Konkurrenz. Eine erste Eingrenzung der Genres durch oben genannte Analysen ist so möglich.

Die aufgeführten Anforderungen an Expressivität, Effektivität und Angemessenheit werden durch ein explizites Baumdiagramm gut erfüllt. So ist eine zweifelsfreie Darstellung der Beziehungen und Abbildung der Hierarchien möglich, ohne unpassende Assoziationen zu wecken. Alle Informationen, die hier für die obere visuelle Ebene der Informationen nötig sind, werden übermittelt. In den Daten versteckte Beziehungen werden offengelegt und konkrete Details durch die Anzeige der Knotenbezeichnungen beim Hovern mit der Maus aufgeführt. Durch das intuitive,

schnelle Verständnis ohne notwendige Erklärungen und Legenden, die überblicksartige Ansicht und die weite Verbreitung solcher Diagramme in vielen unterschiedlichen Bereichen ist auch die Effektivität der Visualisierung hinreichend gegeben. Die durch Hovern angezeigten expliziten Informationen sowie die schnelle quantitative Übersicht über Angebote durch Zählung der von einem Elternknoten ausgehenden Pfade trägt unter anderem dazu bei. Weiterhin ist diese Visualisierungstechnik durch den eher als gering einzuschätzenden monetären und zeitlichen Aufwand zur Erstellung bei starker Expressivität und Effektivität angemessen.

Wie erwähnt, wird durch das explizite Baumdiagramm die Anforderung an die Darstellung von hierarchischen Beziehungen erfüllt. Durch den Einsatz des Walker-Algorithmus wird die Effektivität dank einheitlicher Positionierung der Knoten nochmals verbessert.[17, 688ff.] Vermindert wird sie gleichzeitig durch die hohe Anzahl an Knoten, die zu einem breiteren, leicht weniger übersichtlichen Diagramm führen. Durch das Duplizieren des Diagramms mit veränderter Größe wird der Nachteil der größeren Menge an Knoten reduziert.

Hyperbolische Bäume sind eine Alternative zum expliziten Baumdiagramm. Durch die Zeichnung der Beziehungen und Knoten im hyperbolischen Raum und einen wachsenden Kreisumfang bietet diese Technik mehr Platz für größere Hierarchien.[9, 2f.] Jedoch zöge diese Alternative einen höheren (Kosten-)Aufwand mit sich, weniger Intuitivität in der Erkennung der eigenen und konkurrierenden Angebote sowie eine suboptimale quantitative Übersicht der qualitativen Daten. Implizite Baumdiagramme sind eine weitere Alternative, bei der die Eltern-Kind-Beziehung durch Umschließung des Kindknotens durch den Elternknoten visualisiert werde.[13, S. 394] Sie sind platzsparender, aber durch einen geringeren Bekanntheitsgrad in der Zielgruppe weniger intuitiv verstehbar. Als Überblick bieten sie zudem weniger schnell eine Übersicht über eigene und konkurrierende Angebote statt eines Entlangfahrens der Kanten mit dem Auge.

3.3.2 Visualisierung Zwei

Für die erste Teilbearbeitung der zweiten Anwendungsaufgabe wird der Parallele Koordinaten Plot gewählt, da er sich gut für Visualisierungen von Datenverteilungen und Abhängigkeiten zwischen Attributen eigne.[3, S. 11] In positiver Richtung der y-Achse werden parallel zueinander die in ihrem Wertebereich linear skalierten Achsen positioniert. Ein Datenpunkt hat verschiedene Koordinaten, mittels derer Polygonzüge konstruiert werden. Die Schnittpunkte mit den jeweiligen Achsen zeigen den Attributwert.[8, 25f.] [3, S. 11]

Auf den parallelen Achsen werden die Wertebereiche der Verkaufszahlen der fünf verschiedenen Regionen skaliert und nebeneinander angeordnet abgebildet. Die Schnittpunkte der Polygonzüge, resp. Datenpunkte, mit den Achsen kodieren die jeweiligen Attributausprägungen in den Regionen. Mehrdimensionalität und ein ganzheitlicher Blick werden erfüllt.

Es können sowohl die expliziten Details zu Videospielen und ihren Verkaufszahlen als auch die in den Daten versteckten Informationen zu Muster- und Clusterbildung, Abhängigkeiten zwischen mehreren Regionen und Ausreißer offengelegt werden. Ersteres wird durch die Anzeige der Informationen bei Hovern über ein Polygonzug über dem Plot visualisiert. Zweiteres durch

die Verbindung der Attributausprägungen eines Titels durch Polygonzüge sowie die sich durch Überschneidungen, Überlappungen und Ansammlungen der Linien bildenden Muster. Weiterhin eignet sich die Technik durch die Abbildung der Attributwerte auf Position und in gewissem Maße Orientierung gut für quantitative Daten.[1] Durch eine freie Wahl der Achsenbelegungen mittels Drop-Down-Menü aller Achsen können zudem weitere interessante Muster, um den gesamten erwünschten Blick von oben zu erhalten, erkannt werden. Das Problem der laut Lehman bestehenden Darstellung von Abhängigkeiten nur zwischen drei benachbarten Achsen kann somit umgangen werden.[10, S. 594] Die Expressivität wird insgesamt gut erfüllt.

Muster, Cluster und Ausreißer lassen sich grundlegend gut durch die Polygonzüge erkennen. Nichtsdestotrotz wird die Effektivität leicht gemindert, da das korrekte Verständnis der Kodierung der Daten einigen Personen der Zielgruppe schwieriger fallen kann. Umso wichtiger ist die zuvor beschriebene Möglichkeit zur Wahl der Achsenbelegung sowie prägnanter Erklärtexte. Bei großen Datensätzen ist es in Parallelen Koordinaten Plots problematisch, einzelne Polygonzüge durch Überlagerungen zu verfolgen. Durch ein Hervorheben der Linie und der Informationsanzeige beim Hovern kann dies gelöst werden. Gleichzeitig erhalten alle Polygonzüge eine grüne Färbung mit geringerer Opazität, sodass Schnittstellen und Häufungen von Liniensegmenten durch einen kräftigeren Ton leicht erkennbar sind. Diese Art von Röntgenstrahlentechnik kann die Erkennung von Mustern und Clustern verbessern und verbessert die Effektivität wieder. Das Design der Visualisierung mittels Filterung und freier Wahl der Achsenbelegung verhindert zwar nicht die vielen Kombinationsmöglichkeiten, jedoch die dabei mittels einfachen Achsentauses auftretende Überforderung. Parallele Koordinaten Plots sind zwar etwas (kosten-)aufwendiger zu erstellen, bleiben aber trotzdem im Kontext der Anwendungsaufgabe angemessen.

Eine Alternativen zu Parallelen Koordinaten Plots stellen Icon-Techniken, bspw. Star Plots oder Stickfigures dar. In Star Plots seien die Dimensionen als eckige Achsen ausgehend von einem Kreismittelpunkt mit einer äußeren Verbindungslinie zur Repräsentation der Attributausprägung jeder der Dimensionen visualisiert. Dabei steigt jedoch die Anzahl der zu erstellenden Icons mit steigender Menge an Datenpunkten stark, da sie nur je einen Datenpunkt repräsentieren können.[3, S. 20] Ein ähnliches Problem existiert auch bei Stickfigures, die zusätzlich durch zwei Attribute in einem Koordinatensystem mit weiteren positioniert werden. Zwar können so grobe Muster und Cluster erkannt werden, die anderen Anforderungen bspw. konkrete Informationen zu Datenpunkten werden allerdings unzureichend erfüllt. Intuitivität und Effektivität sind im vorherrschenden Kontext unzureichend, womit Icon-Techniken alternativ ungeeignet sind.

3.3.3 Visualisierung Drei

Die vollständige Beantwortung der zweiten Anwendungsaufgabe bedingt eine weitere Visualisierung, für die die Technik des Scatterplots gewählt wird. In einem kartesischen Koordinatensystem mit orthogonaler x- und y-Achse werden x- und y-Werte zweier variabler Attribute eines Datenpunkts zu bivariaten Paaren zusammengefasst. Durch die Abtragung und Positionierung dieser als Punkte im Koordinatensystem werden die Attributwerte visualisiert.[6, S. 103] Durch

Abbildung vieler Datenpunkte wird die Form von Abhängigkeiten, Muster und Ausreißer zwischen zwei Attributen sichtbar.[3, S. 9] [6, S. 103] Die Schwäche des Scatterplots durch Verlust an Informationen von mehreren Dimensionen wandelt sich in diesem Fall durch eine Detaillierung und Überprüfung der Erkenntnisse des Parallelen Koordinaten Plots zum Vorteil.[3, S. 9] [18, S. 93]

Im Design dieses Scatterplots ist die Belegung der Achsen zur Darstellung der Wertebereiche der Attribute *North America*, *Europe*, *Japan*, *Rest of World* und *Global* frei wählbar, um den Anforderungen nach Auswahl der interessanten Attribute des Parallelen Koordinaten Plots zur Detaillierung dieses nachzukommen. Zudem sind alle möglichen Abhängigkeiten zwischen Attributkombinationen untersuchbar. Die Datenpunkte werden zur visuellen Ablesbarkeit wie oben beschrieben als Attributwert-Kombinationen kodiert, an den Achsen skaliert und als klassische Kreise im Raum des Koordinatensystems positioniert. Quantitative Daten lassen sich am besten auf Position und untergeordnet auch auf Orientierung abbilden, was der Scatterplot bestens umsetzt.[1]

Die Anforderungen an die Expressivität auch in Bezug auf die obere visuelle Ebene der Informationen werden durch die unmissverständliche Zuweisung der Videospielinformationen zum Punkt im Diagramm mittels Hovern sowie die Visualisierung von versteckten Informationen erfüllt. Letztere bestehen aus dem deutlichen Aufzeigen der Art der Korrelationen, Clustern und Ausreißern zwischen zwei wählbaren Attributen. Weiterhin lässt sich mit ihm die gewünschte Vergrößerung der multidimensionalen Daten auf eine Ansicht von nur zwei Attributen und der damit verbundene höhere Detaillierungsgrad umsetzen. Die Analyse des komplizierteren Parallelen Koordinaten Plots kann überprüft und Fehler erkannt werden. Detailfragen zur finalen Entscheidung für ein weiter zu fokussierendes Genre oder eventuelle Korrelationen zwischen Regionen in diesem, die hinder- oder förderlich sein können, werden beantwortet. Durch eine deutliche räumliche Abgrenzung von Punkthaufen jeder Form können Cluster erkannt, bestätigt und einzelne Ausreißer bestimmt werden. Diese konkreten Anforderungen werden damit erfüllt.

Effektivität und Angemessenheit sind klar gegeben. Ersteres ergibt sich aus der starken Verbreitung der Technik in allen Bereichen und der unmissverständlichen, sich durch die Positionierung der Punkte ergebenden Darstellung. Damit visualisiert der Scatterplot intuitiv und schnell lesbar eine Fülle an detaillierten statistischen Informationen. Er ist durch seine Trivialität in der Abbildung zweier quantitativer Attribute angemessen. Bei hoher Expressivität und Effektivität ist der Scatterplot vergleichsweise unkompliziert und kostengünstig..

In diesem Kontext sind keine sinnvollen Alternativen möglich. Korrelationstabellen listen Abhängigkeiten nur tabellarisch auf, visualisieren sie jedoch nicht mittels einer spezifischen Technik. In QQ-Plots können zwei statistische Variablen miteinander verglichen werden, allerdings durch geordnete Gegenüberstellung ihrer Quantile zur Erkennung von Verteilungen. Dies dient nicht der Lösung des Zielproblems, könnte aber in einer zu beauftragenden detaillierteren Marktstudie Verwendung finden. So sei der Scatterplot "the most versatile, polymorphic, and generally use-

ful"[6, S. 103] Visualisierung für statistische Daten und eignet sich perfekt zur abschließenden Beantwortung der zweiten Anwendungsaufgabe und des gesamten Zielproblems.

3.4 Interaktion

In jeder Visualisierung sind Interaktionen mittels Hovern über die Datenpunkte zur Anzeige konkreter Informationen im jeweiligen Kontext des Plots sowie zum Aufleuchten des Datenpunkts zur besseren Erkennung möglich. So werden die Anforderungen an stets mögliche Detailansichten sowie Verbesserungen von Expressivität und Effektivität wie zuvor beschrieben erfüllt.

In jeder einzelnen Visualisierung sind weitere Interaktionsmöglichkeiten integriert. So kann im expliziten Baumdiagramm der vergrößerte Baum mittels Scrollbar verschoben und mithilfe eines Links zu den detaillierteren Visualisierungen gewechselt werden. Von jeder der detaillierten Plots kann der Anwender durch einen Link am unteren Ende der Website schnell einen weiteren Tab zur Rückkehr zum Baumdiagramm als Ausgangspunkt und Übersicht für bspw. einen Abgleich öffnen. Im Parallelen Koordinaten Plot sind die Achsen dank eines Drop-Down-Menüs für jede frei belegbar. Wie in Kapitel 3.3.2 beschrieben, kann dies Überforderung, Unübersichtlichkeit und Zeitverlust durch bloßen Achsentauch verhindern sowie die teils versteckten Abhängigkeiten zwischen den Regionen in jeder Kombination sichtbar machen. Die Betrachtung der Abhängigkeiten zwischen nur drei benachbarten Dimensionen ist so weniger problematisch. Auch der Scatterplot bietet die Interaktion mit je einem Button als Drop-Down-Menü zur freien Achsenbelegung, um detaillierte Untersuchungen von Abhängigkeiten und Vergleiche zwischen zwei beliebigen Attributen zu ermöglichen. Dies dient zudem der gewünschten Vergrößerung und Detaillierung der vorherigen Visualisierung.

Ist der Anwender bei den detaillierten Plots angelangt, kann er mit einem Button als Drop-Down-Menü interagieren und mit diesem leicht und schnell zwischen allen Visualisierungen wechseln. Bei einem Wechsel zum Baumdiagramm leitet der dort platzierte Link zur eigentlichen Visualisierung dieses weiter. Für eine verbesserte Effektivität, Zeitersparnis und Komfort bleiben die Achsenbelegungen bei einem Plotwechsel mittels Drop-Down-Menü gespeichert und müssen bei erneutem Wechsel zurück nicht neu eingestellt werden. Welche Attribute auf den Achsen ausgewählt sind, ist durch eine sich anpassende Unterüberschrift des Plots gekennzeichnet.

Zuletzt wird die Anforderung an einen Filter zur Auswahl der Genres und entsprechender Eingrenzung der Stichprobe in den Visualisierungen Paralleler Koordinaten Plot und Scatterplot umgesetzt. Hierzu gibt es bei beiden Visualisierungen die Möglichkeit zur Interaktion mit einem Drop-Down-Menü, in dem das gewünschte zu betrachtende Genre unter Anpassung der visualisierten Daten ausgewählt werden kann. Die Filterung bleibt bei Umschalten des Plots bestehen und wird direkt auf den anderen angewandt, sodass eine Interaktion mit einem Plot zur Änderung des anderen führt. Durch die Anzeige des Menüs sowie die sich anpassende Unterüberschrift des Plots ist das gewählte Genre nachvollziehbar. Die Zielgruppe kann durch die Interaktion mittels Genrefilter die Spiele und die Abhängigkeiten derer zwischen den Regionen sinnvoller vergleichen und vielversprechende Genres und in diesen möglicherweise Regionen für

die Investition in weitere Marktstudien eingrenzen.

4 Implementierung

Im folgenden Kapitel wird die Implementierung der zuvor beschriebenen Visualisierungen und Interaktionen beschrieben. Der Code wird in fünf Elm-Module unterteilt, bestehend aus je einem Modul für die drei einzelnen Visualisierungen, einem für Datenstrukturen und einem zur Zusammensetzung und Ausführung des Hauptprogramms. *TreeHierarchy* dient als Ausgangspunkt der Anwendung, wobei *MainScatterParallel* den Hauptteil der Anwendung durch die eigentliche Verbindung der Visualisierungen enthält. Innerhalb der Visualisierungsmodule werden zuerst kleinere Funktionen definiert, die im späteren Verlauf genutzt werden, bis allgemeine Definitionen der Plots schließen. Über alle Module hinweg werden diverse Bibliotheken genutzt, die zu Beginn jedes Moduls importiert werden.

Insgesamt zeigte sich die Implementierung der Anwendung als zeitaufwendig, da trotz der umfangreichen Übungsaufgaben die Routine und teils das Verständnis im Programmieren mit Elm fehlte. Die Übungsaufgaben waren jedoch enorm hilfreich und dienten oft als Grundlage der Funktionen. Im gesamten Projekt gab es neue Erkenntnisse und Verständnis für die Funktionen der Sprache und des Quellcodes, sodass sich ein zusätzlicher großer Lerneffekt einstellte. Trotz der guten Vorlagen der Abgaben der Autorin zu den Übungsserien traten einige Schwierigkeiten auf, die in den folgenden Unterkapiteln Erwähnung finden.

4.1 Data.elm

Dieses Modul dient der Deklaration nahezu aller verwendeten Datentypen sowie der Implementierung der Decodierungen. Einzig *type Model* und *type Msg* sind aus Gründen der Übersichtlichkeit und Verständlichkeit des Codes in *MainScatterParallel* dort nicht vertreten.

Zunächst sind die Funktionen *decodeGameSales*, *csvString_to_data* und *gamesSalesList* zur Decodierung der CSV-Dateien implementiert. Die einzelnen Felder der CSV-Datei werden als String decodiert und in eine Liste vom Typ *GameSales* geschrieben. Darauf folgt die Decodierung der JSON-Datei mittels *treeDecoder*. Beide Decoder finden im *update* in *MainScatterParallel* zur initialen Überschreibung der Daten ihre Anwendung. Sie basieren den für die Übungen sieben und neun verwendeten Decodern und wurden an die hier verwendeten Daten angepasst.

Es folgt die Deklaration jener Datentypen, die für alle Visualisierungen außer die *TreeHierarchy* wichtig sind. *Type alias GameSales* definiert alle Daten der CSV-Datei wie sie durch den Decoder decodiert werden und später im Model in *MainScatterParallel* weiterverwendet werden. Die Deklarationen *type RegionType* und *type PlotType* als CustomType werden für die Auswahl der Achsen in den Visualisierungen zwei und drei sowie die Auswahl der anzuzeigenden Visualisierung benötigt. Sie werden im *type Model* und im *type Msg* in *MainScatterParallel*, im *regionFilter* in *Scatterplot* und im *multiDimenData* in *ParallelPlot* verwendet. Es folgen die durch die Übungsserien bekannten Datentypen für die spezifischen Visualisierungen.

Zuletzt werden die für die Buttons wichtigen Funktionen zur Typkonversion von einem String zum jeweiligen CustomType und mit Ausnahme vom PlotType umgekehrt beschrieben. Dies ist nicht nur zur Erstellung von Buttons, die nur Strings erlauben, nötig, sondern auch zur bspw. korrekten Anzeige von aktuell ausgewählten Regionen in der Folge. Weiterhin ist eine Konversion von einem RegionType zu einem Tupel aus GameSales und Float für die Funktion *multiDimenData* in ParallelPlot notwendig.

4.2 TreeHierarchy.elm

Für das Modul können große Teile des von der Autorin in Übungsserie neun erstellten Quellcodes verwendet werden und müssen lediglich minimal auf die verwendeten Daten angepasst werden. Begonnen wird mit den grundlegenden Funktionen *main*, *init* und *update*. Die Anwendung wird initialisiert, die Daten unter Anwendung des Decoders für JSON-Dateien geladen und die jeweils auszuführenden Aktionen bei korrektem oder inkorrektem Laden der Daten bestimmt. Es folgt die Helferfunktion *convert* zur Konvertierung des Baumes basierend auf den Vorgaben zu Übung neun.

Line und *point* definieren die Linie zum Zeichnen der Verbindungen von Eltern zu Kindknoten bzw. zur Darstellung der Knoten an sich sowie die um 75° rotierte, fett gedruckte Beschriftung.

TreePlot2 zur Zeichnung des Baumes ohne Scrollen und *treePlot* zur näheren Ansicht des Baumes werden folgend implementiert. Sie unterscheiden sich einzig in der prozentualen Vergrößerung der global definierten Höhen und Breiten um 200 bzw. 150 Prozent im *treePlot*. Auch die Plots an sich basierend stark auf dem Quellcode der Autorin zu Übung neun. Mittels lokaler Funktionen wird das Layout des Baumes bestimmt, bevor die Abhängigkeiten des Baumes mit Hilfe von *Dict.get* zum Erhalt der X- und Y-Werte berechnet werden, um die Pfade zwischen Eltern- und Kindknoten zu zeichnen. Die Funktionen *checkRootNegative* und *nodeValuePath* verhindern das fehlerhafte Zeichnen einer Linie zum nicht existenten Elternknoten der Wurzel. Nachfolgend werden die berechneten Verbindungen mittels Anwendung zuvor beschriebener Funktion *line* auf *nodeValuePath* gezeichnet. Berechnete Knoten werden durch Anwendung von *point* auf *nodeValue* gezeichnet. Die Definition des Designs inklusive Hovern wird im global definierten CSS *cssTree* vorgenommen.

In der *view*-Funktion werden lokal die Daten in einen Baum konvertiert, bevor dies neben dem CSS den beiden Zeichnungsfunktionen des Baumes übergeben und diese dann ausgegeben werden können. Die Anordnung sowie der Designstil der HTML-Elemente ist an *MainScatterParallel* angepasst.

Zuletzt werden global allgemeine konstante Einstellungen zum Plot definiert. Diese Definitionen basieren mit Unterschied bestimmter Werte auf den Vorgaben in der Vorlage zum Arbeiten mit Bäumen aus Übung neun und den Abgaben der Autorin zu Übung neun.

Die Implementierung der expliziten Baumhierarchie bereitete keine Schwierigkeiten, da sehr viel auf den abgegebenen Codes der Autorin zu Übung neun basiert. Lediglich der Decoder in *Data* musste an die Datenfelder angepasst, der Link zum Laden der Daten geändert sowie

die besten Design- und Darstellungsweisen inklusive der zweifachen Darstellung des Baumes gefunden werden. Das CSS wurde gegensätzlich global definiert und das allgemeine Design Seite im *view* an jenes aus *MainScatterParallel* angepasst.

4.3 ParallelPlot.elm

Dieses Modul beginnt nach dem Import mit drei Funktionen zur Zuordnung der Daten zum benötigten Datentyp und der Reduzierung der Datensätze um solche, die Null-Values enthalten. Letzteres ist nach der manuellen Vorverarbeitung der Daten zwar nicht zwingend notwendig, jedoch kann so auf Fehler und Änderungen der Rohdaten eingegangen werden.

Die erste Hilfsfunktion dient mittels *Maybe.map2* dem Pipen. Basierend auf einer normalen *Maybe.map5* Funktion wird die zweite Hilfsfunktion um einen Parameter erweitert. Beide begründen sich aus den Dokumentationen der Elm-Core-Bibliothek zu *Maybe*.^[4] Da es kein *Maybe.map6* gibt, wird in *helpMapBig* die vorherige Hilfsfunktion angewandt, um dieses Problem zu umgehen. *AssignmentAndReduce* wiederum wandelt schließlich lokal mithilfe der *helpMapBig* die Daten vom Typ *GameSales* zu *Maybe GameSales* um und reduziert diese um die Datensätze mit fehlenden Werten mittels *List.filterMap*. Diese Funktion basiert auf den Abgaben der Autorin zu Übung sechs.

Die Funktion *multiDimenData* dient zusammen mit der lokalen Funktion *multiDimFunction* aus *MainScatterParallel* der freien Auswahl und Belegung der Achsen sowie der Anwendung der Filterung nach Genre. Das Grundgerüst beider Funktionen basiert auf den Abgaben der Autorin zu Übung sechs. Der dort implementierte Achsentausch wurde aufgrund der Anforderungen in diesem Projekt geändert. *MultiDimenData* benötigt als Eingabe eine Liste von *GameSales* sowie fünf *RegionTypes* als die auszuwählenden Regionen für die Achsen. Weiterhin braucht es den Namen des Spieles und den Publisher des Spieles zur Anzeige dieser im späteren Plot. Der Typ begründet sich aus dem *type alias GameSales*. Zusätzlich sind fünf Namen, also Beschriftungen als *String* nötig. Begründet ist dies in der Definition der Recordfelder in *type Model* sowie dem für das *update* benötigten Datentypen (*RegionType*, *String*). Die Funktion muss den Typ *MultiDimData* zur Übergabe an *scatterplotParallel* und Zeichnung des Plots ausgeben. Eine besondere Schwierigkeit bei der Konstruktion der Funktion lag im Umfang dieser und im Verständnis des benötigten Datentyps. So muss der *RegionType* innerhalb der Funktion mittels *regionTypeToAxisAnnotation* nochmals umgeschrieben werden. Die Funktion wird in *MainScatterParallel* in der lokalen Funktion *multiDimFunction* angewandt und bekommt dort die von ihr benötigten Daten übergeben.

Es folgt der *scatterplotParallel*, der nahezu unverändert aus den Abgaben der Autorin zu Übung sechs übernommen werden kann. Es werden unter anderem mittels lokaler Funktionen die Achsen in X-Richtung positioniert sowie folgend inklusive Beschriftungen in einem umgebenden Rechteck gezeichnet. Die Zeichnung der Datenpunkte wird durch *Shape.line* und *Shape.lineaerCurve* realisiert. Eine Änderung ist der beim Hovern über einen Datenpunkt anzuzeigende Text. Er wird lokal beschrieben, bevor dieser Funktion zuletzt die entsprechenden Daten

übergeben werden.

Auch hier wird ein globales *cssParallel* definiert. Zur Entstehung des gewünschten Röntgen-effektes zur erleichterten Identifikation der Cluster und Auffälligkeiten der Datenpunkte wird die Opacity der Datenpunkte auf 0.5 festgelegt. Durch das Einfärben in einem stärkeren Grün-ton beim Hovern und dem Einblenden der Details zum Datenpunkt lassen sich die einzelnen Videospiele besser nachvollziehen.

Die Implementierung schließt mit den generellen Einstellungen für den Plot, welche mit Aus-nahme bestimmter Werte auf den Abgaben der Autorin zu Übung sechs basieren.

Die Erstellung des ParallelPlot stützte sich in weiten Bereichen auf den Abgaben zu Übung sechs. So bereitete das Zeichnen des Plots wenig Schwierigkeiten. Etwas komplizierter war zu-nächst die Implementierung des Mappings und der Filterung nach Null-Werten durch das fehlen-de *List.map6*. Schwieriger gestaltete sich jedoch die Umsetzung der freien Auswahl der Achsen wie zuvor beschrieben. Hier war viel Ausprobieren und Verständnis, welcher Datentyp für wel-che weitere Funktion benötigt wird und welche durch die vorherige Definition der Datentypen vorhanden waren, nötig.

4.4 Scatterplot.elm

Das Modul enthält ähnliche drei Funktionen mit demselben Zweck wie *ParallelPlot*. Die ersten beiden Funktionen gleichen sich. *FilterAndReduceGames* basiert jedoch auf den Abgaben der Autorin zu Übung eins. Die Daten werden zunächst mithilfe von *helpMapBig* vom Typ *GameSales* dem Typ *Maybe Point* zugeordnet. Die zugeordneten und mit *List.filterMap* gefilterten Daten in *filter* werden dann unter anderem in *XyData* geschrieben, dass als Output der Funktion *FilterAndReduceGames* definiert ist. *XyData* werden benötigt für das Zeichnen des Scatterplots, da sie unter anderem eine Liste Daten des Typs *Point* enthalten.

Die Funktion *regionFilter*, basierend auf den Abgaben der Autorin zu Übung vier, filtert mit Eingabe einer Liste *GameSales* und des *RegionTypes* die Datenpunkte nach den je gewünsch-ten Regionsauswahlen für die Achsen. Sie wird zur Interaktion innerhalb des Plots, zur freien Auswahl der Achsen und korrekten Anzeige der entsprechenden Datenpunkte benötigt. Die An-wendung wird in *MainScatterParallel* vorgenommen und beschrieben.

In *point* wird die Positionierung der Punkte sowie deren Beschriftung für die Weiterverwen-dung in *Scatterplot* festgelegt. Der *scatterplot* ist für das Zeichnen des Plots zuständig. Mit der lokalen Funktion *pointsXY* wird die Variabilität der Achsen und damit X- und Y-Werte der Punkte ermöglicht. Im SVG des *scatterplot* wird die Positionierung der X- und Y-Achse, ihrer Beschriftungen und der Punkte festgelegt. Letzteres wird durch die lokalen Funktionen *xScale-Local*, *yScaleLocal* und *pointsXY* sowie der globalen Funktion *point* erreicht. Der *scatterplot* basiert auf den Abgaben der Autorin zu Übung 1.4 sowie Übung vier für die lokale Funktion *pointsXY*.

Global wird wiederum das *cssPoint* definiert. Zuletzt werden wie zuvor allgemeine, konstant für den Plot geltende Einstellungen, basierend auf Übung eins bis vier, global definiert.

Die Implementierung des Scatterplots verlief in weiten Teilen problemlos, wenn auch langsam durch ständiges Überprüfen der richtigen Datentypen. Die Umsetzung des Mappings und Filters nach Null-Werten stellte wie beim *ParallelPlot* ein Problem dar, wobei die Lösung vereinfachend größtenteils von dort übernommen werden konnte. Weiterhin musste beachtet werden, dass die Achsen des Plots frei wählbar, also variabel definiert sind, anders als in den meisten Übungsreihen. Mit entsprechender Zeit- und Denkinvestition war auch dies überwindbar.

4.5 MainScatterParallel.elm

Auch das Modul *MainScatterParallel* definiert zunächst die Importe. Aufgrund der besseren Erkennbarkeit des Ursprungs von Funktionen, werden sie nicht dezidiert importiert, sondern nur das allgemeine Modul.

Es wird die gesamte Elm-Architektur verwendet, da das Modul den *Scatterplot* und die *ParallelPlot* ausführen und alle drei Visualisierungen verbinden soll. Hierzu wird das Programm zunächst wieder in der *main*-Funktion definiert sowie die *subscriptions*- und die *init*-Funktion implementiert.

Danach folgt die Definition des *type Model* und *type Msg*. In *type Model* werden die drei Varianten Error, Loading und Success beschrieben, die in den entsprechenden Fällen angezeigt werden. Die Variante Success ist gleichzeitig ein Record aus den dort beschriebenen Feldern. Wichtig zu beachten ist, dass die initialen Werte erst in der *update*-Funktion hier hinein geschrieben werden. Im Feld *data* können die geladenen und decodierten Daten als Liste vom Typ *GameSales* gespeichert werden. Zur Ermöglichung der Interaktionen gibt es weiterhin Felder für das Genre, die Achsen eins bis fünf, die Namen der Achsen eins bis fünf, die X- und Y-Achse und den gewünschten Plot. Für die Achsen des Scatterplots und Parallelen Koordinaten Plots sowie den Plot an sich sind die entsprechenden CustomTypes definiert. Der *type Msg* definiert die Varianten für die Nutzung in *update* zur Überschreibung des Models. Es gibt eine Variante zum erfolgreichen Laden der Daten und damit der Initialisierung des Plots sowie für jeden möglichen Wechsel von Achsen in den Plots sowie den Wechsel des Plots an sich.

In der *update*-Funktion werden zuerst die initial anzuzeigenden Attributwerte mit Rücksicht auf die zuvor festgelegten Datentypen definiert und nach Anwendung der Decodierung in Success gespeichert. Weiterhin werden für alle Interaktionen mit dem Model die Varianten von *Msg* eingefügt und beschrieben, wie bei einer Änderung dieser Variante durch Klick auf deinen Button das Model überschrieben wird.

Nachfolgend werden alle benötigten Button-Funktionen zur Überschreibung des Models bei Klick auf eine Variante implementiert. Wichtig ist die Nutzung der in *Data* definierten Umwandlungsfunktionen für alle außer den *buttonGenreType* von Strings, welche die Buttons benötigen, zu den entsprechend für das Model definierten Datentypen. Die Änderungen können dann in der Variante des *Msg*-Typen gespeichert und durch die *update*-Funktion das Model überschrieben werden. Die Buttons basieren auf den Abgaben der Autorin aus Übung vier.

Die Funktion *filterGenre* beruht auf den Abgaben der Autorin zu Übung vier und wird folgend

im *view* angewandt. Dem Anwender wird dadurch eine Filterung nach dem Attribut Genre ermöglicht.

In der *view*-Funktion werden die verschiedenen Fälle des Models zur Ansicht behandelt, wobei hier nur auf den Erfolgsfall näher eingegangen wird. Zunächst wird eine *let-in*-Konstruktion auf erster äußerer Ebene erstellt. In lokalen Funktionen wird *gameSalesData* sowie die Anzahl der Spiele insgesamt beschrieben. Zusätzlich wird hier der globale *filterGenre* zur Filterung der Daten nach gewünschtem Genre angewandt. Auch die Länge dieser Liste wird berechnet. Damit nun zwischen den Plots gewechselt werden kann, werden die verschiedenen Varianten bzw. Fälle des Datentyps *PlotType* wie im Record der Variante *Success* des Models aufgeführt im *in*-Teil angelegt. Je nachdem, welcher Plot mittels Button ausgewählt wird und wie das Model durch das *update* überschrieben wurde, wird einer der drei Fälle angezeigt. Der Zustand des Models mit dem gefilterten Genre soll bestehen bleiben, sodass eine Interaktion mit dem Genrebutton in einer Visualisierung auch in der anderen angewandt und übernommen wird. Dazu dient zum einen die zuvor beschriebene Variantendefinition im *in*-Teil der äußeren *let-in*-Ebenenkonstruktion sowie jeweils eine weitere *let-in*-Konstruktion in den Falldefinitionen. Diese befinden sich auf zweiter innerer Ebene, sodass die in der äußeren Konstruktion berechneten Zustände, sprich Genreauswahlen, auch hier gelten und bei einem Wechsel des Plots übernommen werden. In den lokalen Funktionen der inneren Konstruktion der Plots muss dazu die lokale *gameSalesDataFiltered* der äußeren Ebene angewandt werden. Würde in jeder der inneren Ebenen für jeden Plot die Filter nach Genre neu definiert werden und lokal angewandt statt in der übergeordneten Ebene, würden die Einstellungen respektive der Zustand des Models nicht übernommen werden.

Im ersten Case wird der ParallelPlot behandelt. Lokal wird die *assignmentAndReduce*-Funktion auf die gesamten Daten sowie auf die in der äußeren Ebene nach Genre gefilterten Daten angewandt. In *multiDimFunction* wird *multiDimenData* letztlich durch Übergabe der entsprechenden bereinigten und gefilterten Daten genutzt. Im *in*-Teil der inneren Ebene wird die Struktur und das Design der Seite festgelegt. Für eine angenehmere und wenig ablenkende Ansicht werden die Farben den Grüntönen in den Plots angepasst. Zur Direktion des Auges über die Seite und verbesserten Erkennung von wichtigen Abschnitten wie der Auswahl der Plots, des Genres und der Achsen werden Ränder und verschiedene Grüntonabstufungen genutzt. Neben Informationstexten und Überschriften werden hier auch die Buttons als Dropdown-Menü zum Wechsel des Plots, zur Änderung des Genrefilters sowie der Anpassung der Achsen eingefügt. Eine transparente Anzeige der gewählten Einstellungen und verbleibenden Videospiele wird hinzugefügt. Schließlich wird dem *scatterplotParallel* das *cssParallel* sowie zwei Konstanten für die Höhe und die Aspect Ratio übergeben, bevor die zuvor lokal berechnete *multiDimFunction* folgt. Zuletzt wird noch ein Text mit Link direkt zurück zur TreeHierarchy in einem neuen Tab angeboten.

Im zweiten Case wird der Scatterplot behandelt. Das Vorgehen mit den lokalen Funktionen zur Anwendung der Filter und des Mappings ist ähnlich wie zuvor. Durch die lokalen Funktionen *valuesX* und *valuesY* werden dem *regionFilter* die nach Genre gefilterten Daten als Liste von GameSales sowie die RegionTypes der X und Y-Achse übergeben. Somit wird der Filter nach

Regionen angewandt und die Interaktion mit freier Auswahl der Achsen kann funktionieren. Der *in*-Teil der Plotvariante Scatterplot ähnelt dem des ParallelPlots stark und unterscheidet sich nur in den Informationstexten sowie der Auswahl der Achsen. Zuletzt wird dem *scatterplot* der *cssPoint*, die zuvor berechneten *gameSalesDataCleared* sowie die *valuesX* und *valuesY* übergeben. Zur Beschriftung der Achsen werden diese mittels der Umrechnung von *RegionType* zu String als String hinzugefügt.

Da die *TreeHierarchy* in *TreeHierarchy.elm* implementiert ist, muss für diese Plotvariante nur das *Html* inklusive Link zum eigentlichen Plot des Baumdiagramms im selben Design wie zuvor definiert werden.

Durch die zuvor in getrennten Anwendungen implementierten Visualisierungen war ein grundlegendes Zusammenfügen dieser durch stückweises Kopieren unkompliziert umsetzbar. Besonders die Erstellung einer einheitlichen *update*-Funktion war jedoch zeitlich durch die Menge an Interaktionsmöglichkeiten aufwendig. Weiterhin schwieriger war die Implementierung der des Dropdown-Menüs für den ParallelPlot, da die Buttons zunächst je einzeln vorlagen. Durch Nachdenken und Ausprobieren vor allem mit der Funktion *multiDimenData* und verschiedenen Konvertierungsfunktionen konnten sie doch wie gewünscht realisiert werden. Zur Überprüfung immer wieder zeitaufwendig das *update* angepasst werden. Zunächst wurden beide Plots als untereinander auf einer Seite gezeichnet implementiert. Entsprechend war ohne Plot-Selektor nur eine *let-in*-Konstruktion nötig und die Filterung wurde für beide übernommen, was nicht dem Gewünschten entsprach. Nach schrittweisem Hinzufügen der Auswahl des Plots und allen zugehörigen Codeteilen, wurde durch Ausprobieren entdeckt, dass eine Übernahme des Modellzustands durch die Schachtelung der *let-in*-Konstruktionen im *view* und der Anwendung des Variantenwechsels für den Plot mittels *PlotType* möglich war. Hier war es im Verlauf kompliziert, den Überblick über die Schachtelungen sowie außen und innen definierten lokalen Funktionen zu behalten.

5 Anwendungsfälle

Im folgenden Kapitel wird ein möglicher Anwendungsfall der implementierten Visualisierungen präsentiert. Dabei präsentiert ein mittlerer Manager des Publishers *505 Games* die Visualisierungen zur Vorstellung der Erkenntnisse dem oberen Management sowie den Stakeholdern des Unternehmens, um weiter analysierend eine fundierte Investitionsentscheidung zu treffen. Es sollen Indikatoren für potenzielle Investitionen in neue Videospiele bzw. Fortsetzungen vorhandener Titel eines Genres gefunden werden, die zu einer Entscheidung zur Inauftraggabe detaillierter Marktstudien in den identifizierten Genres und potenziell zu fokussierenden Regionen führen.

5.1 Anwendung Visualisierung Eins

Wie in allen Visualisierungen übereinstimmend entworfen, wird auch im expliziten Baumdiagramm mittels teils minimaler farblicher Unterschiede der einzelnen Websitebereiche und der

Abgrenzung der Container durch Umrandungen das Auge der Betrachter schnell und indirekt auf die wichtigen Steuerungselemente sowie die eigentliche Visualisierung gelenkt. Sitzt eine Person mit Farbsehschwäche unter den Präsentationsteilnehmern, so ist dies durch Abstufungen der Grüntöne und des Kontrastes auf der Website und in den Visualisierungen unproblematisch.

Der erste Ausschnitt des Baumdiagramms gewährt den Teilnehmern einen Überblick über alle Publisher sowie die eigene Position im Baum. Im zweiten, detaillierteren Ausschnitt wird den Managern und Stakeholdern dargestellt, welche Genres sie mit *505 Games* bedienen und welche ihrer Spiele in welches Genre passen. Durch die einfache Baumstruktur ist schnell erkennbar, dass sie fünfzehn Videospiele in acht Genres anbieten, wobei *Adventure*, *Racing* und *Shooter* die meisten Spiele aufweisen. Dank Einblendung der Titel und Genres beim Hovern über die sich grün färbenden Knoten in beiden Diagrammen sowie das Scrollen im zweiten Diagramm sind konkrete Ein- und Überblicke effektiv möglich. Die Genres *Action* und *Sports* werden aufgrund der bisher eher geringen Ausprägung im eigenen Unternehmen näher betrachtet.

Durch die hohe Zahl der anbietenden Publishern ergibt sich vor allem Konkurrenz im Genre *Action*, wodurch eine Sicht auf mögliche Abhängigkeiten zwischen den Regionen der Welt und die entsprechend beste Positionierung im Markt wichtig ist. Schnell erkennbar durch Struktur und Positionierung der Knoten, zeigen sich vor allem *Activision*, *Capcom*, *Ubisoft* und *Warner Bros.* durch ein hohes Spieleangebot als stärkste Konkurrenz. Das große Angebot in dem Genre kann Potenzial durch möglicherweise viele Käufer bieten. *Sports* hat weniger Konkurrenz, allerdings wird es durch Publisher wie *EA Sports* und *2K Sports* dominiert. Aufgründdessen entschied der Präsentator im Vorfeld, nachfolgend *Action* und *Sports* zu fokussieren.

Auch ein implizites Baumdiagramm hätte dem Informationsbedarf genügt. Aufwand und Schwierigkeit der Implementierung wären im Vergleich leicht erhöht, das Verständnis bei den potenziell ungeschulten Betrachtern wie bspw. Stakeholdern, gerade in kurzen Momenten einer Präsentation, aber gemindert. Hyperbolische Baumdarstellungen wären möglich, würden gut verstanden, zögen aber erhöhten Aufwand und Schwierigkeiten der Implementierung mit sich.

5.2 Anwendung Visualisierung Zwei

Mit der Visualisierung des Parallelen Koordinaten Plots präsentiert der mittlere Manager einen detaillierteren Blick auf die Verkaufszahlen der Videospiele eines Genres in allen Regionen der Welt. Es finden sich Erklärungen zu Einheiten und Darstellungsweisen auf der Website.

Das festgelegte Genre *Action* wird ausgewählt und sodass die Visualisierung mit zunächst voreingestellten Achsenbelegungen und den gefilterten Videospiele betrachtet werden kann. Schnell fällt Japan auf, das viele Titel mit Verkaufszahlen von null aufweist. Weiterhin scheint es durch im Vergleich besser oder schlechter verkaufte Spiele keine Beziehungen zwischen den Regionen mit Japan zu geben. Um Zusammenhänge zwischen den anderen Regionen besser erkennen zu können, wird die letzte Achse unverändert mit Japan belegt (s. Abb. ...). Es lässt sich immer dasselbe Muster erkennen, bestätigt und verdeutlicht durch diverse Kombinationen der Achsenbelegung und der Röntgenstrahlentechnik. Einerseits befindet sich der Großteil der

Videospiele in allen Regionen im unteren Drittel der Verkaufszahlen, was durch die Überlappungen der Datenpunkte und damit kräftigeren Grüntöne gut erkennbar ist. Andererseits scheint es Abhängigkeiten zwischen allen Regionen außer Japan zu geben, erkenntlich durch die konstant waagrecht verlaufenden Datenpunkte. *Grand Theft Auto V* von *Rockstar Games* sticht als Ausreißer heraus.

Durch die positive Korrelation scheint eine möglichst universelle Aufstellung und Nutzung der Synergieeffekte durch Abhängigkeiten nützlich. Positive Korrelationen werden zudem als erstrebenswert für global agierende Unternehmen gedeutet, was das Genre *Action* vorteilhafter für eine weitere Investition macht. Gleichzeitig weist eine Clusterbildung im unteren Drittel der Verkaufszahlen aller Regionen auf starke Konkurrenz oder mögliche Sättigung im Genre hin. Sollte die Entscheidung zu weiteren Marktstudien in diesem Genre getroffen werden, müsste anhand weiterer Einflussparameter unter anderem der Grund für die hohen Verkaufszahlen des Ausreißers geprüft sowie die Sättigung des der Märkte und Einflussfaktoren der Verkaufszahlen der weiteren Spiele analysiert werden. Aufgrund der unkomplizierten Erkenntnisse aus dieser Visualisierung ist eine Überprüfung im Scatterplot nicht zwingend nötig.

Im Genre *Sports* gibt es keine Ausreißer über alle Regionen hinweg. Auch die Verkaufszahlen verteilen sich stärker über die Skalen. Japan mit in nahezu allen Titeln erkennbaren Verkaufszahlen von null bleibt bestehen und wird aus weiteren Analysen ausgenommen. Im unteren Bild erkennbar ist jedoch die leichte Bildung verschiedener Cluster, die sich auch mit unterschiedlichen Achsenbelegungen bestätigt. Es scheint eine Gruppe an Spieletiteln zu geben, die in Nordamerika und dem Rest der Welt gut verkauft werden konnten, global gesehen und vor allem in Europa deutlich schlechter. Weiterhin deutet sich, wenn auch nur sehr schwach sichtbar, ein Cluster an, in dem Spiele überall gut verkauft werden, in Nordamerika im Vergleich jedoch nicht. Die restlichen Spiele bilden ein weiteres Cluster am unteren Rand der Skala mit durchgängig positiven Abhängigkeiten. Da sich diese Cluster besonders durch Unterschiede zwischen den Verkaufszahlen in Europa und Nordamerika unterteilen, werden diese Regionen im Scatterplot näher betrachtet.

Der Vergleich mehrerer Dimensionen resp. mehrerer Attribute eines Videospiele wäre auch durch Icontechniken möglich. Diese Methode ist jedoch durch die Anzahl der zu erstellenden Icons und Kodierung der Daten auf diese aufwendiger zu erstellen und die Analyse gestaltet sich durch möglicherweise zu wenig vorhandene Datenpunkte zur Musterbildung und einer erschwerten Erkennung konkreter Zusammenhängen zwischen allen Regionen kompliziert.

5.3 Anwendung Visualisierung Drei

In seiner Präsentation wählt der mittlere Manager nun den Scatterplot im Drop-Down-Menü aus. Die zuvor getroffene Auswahl des Genres, zuletzt *Sports*, bleibt bestehen und er belegt Achsen in einem weiteren Drop-Down-Menü mit *North America* und *Europe*.

Die zuvor erkannten Cluster in den Verkaufszahlen im Vergleich von Nordamerika zu Europa verdeutlichen sich. Manager oder Stakeholder, die die vorherige Visualisierung möglicherweise

weniger gut nachvollziehen konnten, verstehen nun zumindest deutlich die Abhängigkeiten zwischen zwei visualisierten Dimensionen. Im unten abgebildeten Scatterplot ist deutlich ein positiv korreliertes Cluster sichtbar. Im zweiten zuvor entdeckten Cluster lässt sich auch bei Achsentausch keine Korrelation zwischen den Regionen finden, kennzeichnet durch die rein waagerechte bzw. senkrechte Verteilung. Einzig das dritte Cluster lässt sich nicht direkt bestätigen, deutet sich aber durch einen Punkthaufen nahe des Ursprungs an. Die Erkenntnisse aus der zweiten Visualisierung werden bestätigt. Sportvideospiele des ersten Clusters, die in Nordamerika gut verkauft wurden, hatten auch in Europa einen guten Absatz und durch Achsentausch erkennbar auch umgekehrt. Dort zeigt ein flacherer Anstieg zeigt auch, dass gut in Europa verkaufte Spiele in Nordamerika weniger gut verkauft wurden. Sehr gute Verkaufszahlen der anderen Gruppe in Nordamerika haben keinen Einfluss auf die Verkäufe in Europa, schlechte Verkäufe in Europa ebenso nicht auf die nordamerikanischen. Durch Einblenden konkreter Details der Datenpunkte wird ersichtlich, dass jene Cluster mutmaßlich in der unterschiedlichen Popularität der Sportarten in den Regionen begründet sind. Fußball scheint mit Abstrichen auch in Nordamerika beliebt, Football in Europa weniger.

Eine sinnvolle Alternative zu Scatterplots ist nicht gegeben, da sie die leichteste, deutlichste Darstellung von Beziehungen zweier Attribute eines Datenpunktes liefern.

Abschließend kann sich das obere Management und wichtigsten Stakeholder von *505 Games* zu einer Investition in eine (kosten-)intensive Marktstudie zur Beurteilung weiterer Einflüsse im Genre *Sports* entscheiden. Die eigene Präsenz dort ist ausbaufähig, es gibt weniger und in der Masse weniger starke Konkurrenz als bei *Action* sowie keine extremen Ausreißer. Zwar verzichtet man auf Nutzung der positiven Korrelationen zwischen allen Regionen, kann aber durch Fokussierung auf beliebte Sportarten des europäischen Marktes diese Abhängigkeiten, begrenzt auch in Nordamerika, nutzen. Für einen vergleichsweise kleineren Publisher ist diese Strategie durch effizienten Einsatz von Mitteln sinnvoller.

6 Verwandte Arbeiten

In diesem Kapitel folgt eine knappe Literatursuche nach ähnlichen Anwendungen zu Videospiegelverkäufen oder Verkaufsdaten im Allgemeinen im Bereich der Informationsvisualisierung und Visual Analytics.

Der erste zu diskutierende Artikel ist *VizInteract: Rapid Data Exploration Through Multi-touch Interaction with Multi-dimensional Visualizations* von Chakraborty und Stuerzlinger.[2] In diesem Artikel wird *VizInteract* vorgestellt und getestet. Es dient der Multi-Touch Interaktion, um multidimensionale Datenvisualisierung schneller und einfacher konstruieren und mit ihr interagieren zu können. Es sind verschiedene Datenvisualisierungen implementiert, die mittels Verschieben und Übereinanderschieben zu neuen Plots resultieren können. So können laut den Autoren bspw. zwei orthogonale Histogramme durch Übereinanderschieben einen Scatterplot kreieren. Forschungsziel des konkreten Artikels die Beobachtung des Nutzerverhaltens der

Anwendung einfacher touchbasierter Interaktionen.

Als Anwendungsfall wird für die Tester unter anderem ein Unterdatensatz der *Video Game Sales 2019* bereitgestellt, mit dem sie Visualisierungen mittels des Tools erstellen und damit vorgefertigte Analysefragen beantworten sollen. Der Datensatz ähnelt dem hier verwendeten, gleicht sich jedoch nicht. In *VizInteract* sind Histogramme, Scatterplots, Parallele Koordinaten Plots, Scatterplotmatrizen und Sterndiagramme möglich. Somit stimmen auch zwei der im Projekt verwendeten Visualisierungstechniken mit denen im Artikel überein.

Gemeinsamkeiten zwischen der Anwendung und Implementierung im vorliegenden Projekt und dem Artikel liegen in der Möglichkeit, Notwendigkeit und Umsetzung von Interaktionen für sinnvolle Visualisierungen, bspw. mittels Filtern. Weiterhin werden in beiden Fällen Scatterplots und Parallele Koordinaten eingesetzt und in den den Testern gestellten Aufgaben ähnlich angewandt wie im Anwendungsfall des Projektes. Durch den Umfang von *VizInteract* und die Tests im Artikel zeigt sich wiederum die Wichtigkeit, die die Autoren den Visual Analytics beimessen. Klar ersichtlich ist die Gemeinsamkeit der Nutzung der Videospiegelverkäufe mit einem sehr ähnlich aufgebauten Datensatz sowie Visualisierungen. Unterschiede ergeben sich aus den weiteren Visualisierungsmöglichkeiten des Tools, der Bedienbarkeit mit Touch-Oberflächen sowie der fortgeschrittenen Interaktion mittels Komposition der Techniken. Im Artikel wird das Tool anders als hier vorliegend mittels eines Anwendungsfalles auf seine Hauptmerkmale geprüft und seine Handhabung fokussiert.

Als Zweites sei *Intelligent Visual Analytics Queries* von Hao et al. genannt.[7] In diesem Artikel möchten die Autoren Analysten mittels ihrer *Intelligent Visual Analytics Query* unterstützen. Der Fokus liegt auf großen multidimensionalen Datensätzen, in die durch das Tool Einsicht in komplexe Muster, Phänomene und Ausreißer erlangt werden soll. Die angestrebte Anwendung beschreibt einen Analysten, der in einer noch unübersichtlichen Visualisierung der Daten einen Interessenbereich sowie die dazugehörigen Attribute auswählt. Anschließend sollen durch automatische analytische und visuelle Analysemethoden Charakteristiken und Beziehungen zu anderen Attributen und Datenpunkten identifiziert werden. Der im Artikel genutzte Anwendungsfall bezieht sich auf Verkaufszahlen von Produkten und Kundenkaufverhalten, da Verkaufsanalysten Produktverkäufe und Promotionen laut der Autoren korrelieren wollen.

Das vorgestellte Tool nutzt eine tabellenartig angelegte visuelle Karte, bei der in den Zeilen die Gruppen an Dimensionen und in den Spalten die Datenintervalle gespeichert werden. Die Farbe kennzeichnet den Attributwert für den Datenpunkt. So können Korrelationen und Ähnlichkeiten mehrerer Attribute erkannt werden. Weiterhin wird wie in der hier vorliegenden Arbeit der Parallele Koordinaten Plot visualisiert. Es können interessante Untergruppen von Daten ausgewählt und die paarweise Korrelation der gewählten Attribute berechnet werden. Die Achsen werden schließlich so angeordnet, dass hochkorrelierte Attribute nah beieinander sind. Auch Scatterplots werden genutzt, wobei diese hier so angeordnet werden, dass weiterhin multidimensionale statt zweidimensionale Daten abgebildet werden können.

Gemeinsamkeiten bestehen in der Auswahl der Visualisierungstechniken und der Anwendung

dieser zum Erkennen von Mustern und Korrelationen in großen multidimensionalen Datensätzen zu Verkaufszahlen. Zusätzlich wird die Nützlichkeit dieser Techniken und der Visual Analytics an sich für die Auswertung und das Verständnis von Verkaufsdaten ähnlich groß eingeschätzt. Auch die Nutzung und Bewertung der Existenz von Interaktionsmöglichkeiten in den Visualisierungen durch Filter deckt sich mit denen der Autorin der vorliegenden Arbeit. Unterschiede befassen sich mit dem Einsatz von Scatterplots für mehr als zwei Dimensionen. Trotz der Ähnlichkeiten der Parallelen Koordinaten Plots ist die Anordnung der Achsen hier verändert und nimmt dem Nutzer Arbeit ab. So muss dieser nicht selbst die Achsen variabel aussuchen, um Muster und Korrelationen besser zu erkennen. Gleichzeitig fehlt ihm diese Flexibilität bei Hao et al. Wie schon im vorherigen Artikel wird weniger die Beantwortung einer Frage aus dem Anwendungsfall heraus behandelt, sondern das Tool allgemein vorgestellt.

7 Zusammenfassung und Ausblick

In dieser Arbeit wurde mittels drei Visualisierungstechniken eine grobgranulare Marktanalyse der Verkaufszahlen des Videospielemarktes für die Konsole *XBoxOne* durchgeführt, um die Zielgruppe des mittleren und oberen Managements sowie eingeschränkt der Stakeholder der Videospielverlage in ihren Investitionsentscheidungen zu unterstützen. Sie erhalten ein schnelles Verständnis und Überblick über den aktuellen Videospielmarkt mit Fokus auf den Verkaufszahlen. Zusammenhänge und Muster zwischen den Regionen in den Genres können ohne Vorwissen im Bereich der Visual Analytics abgelesen, analysiert und präsentiert werden. So werden erste Erkenntnisse bezüglich neuer Möglichkeiten und Chancen im Markt sowie Ansatzpunkte für Strategieentscheidungen ermöglicht. Sie dienen als Entscheidungsunterstützung für die Beauftragung detaillierterer und teurerer Marktstudien für neue Investitionen.

Durch die Implementierung eines expliziten Baumdiagramms konnte eine hierarchische Übersicht über den Markt sowie die Zusammenhänge zwischen Publisher, Genre und Titel des eigenen Verlagshauses und der Konkurrenz umgesetzt werden. Mittels eines Parallelen Koordinaten Plots zur Darstellung mehrdimensionaler Daten konnten auch durch den Einsatz einer Röntgenstrahlentechnik Abhängigkeitsmuster in den Verkaufszahlen von Videospielen eines Genres über mehrere Dimensionen, also Regionen, hinweg erkannt werden. Zuletzt konnten in einem klassischen Scatterplot die Erkenntnisse aus der vorherigen Visualisierung inklusive Clusterbildung, Ausreißern und Korrelationen zwischen zwei Regionen konkretisiert werden. Für ein verbessertes, komfortables und schnelles Anwendungserlebnis sowie Effektivität und Effizienz wurde die Auswahl der Genres mittels Drop-Down Menü interaktiv gestaltet sowie beim komfortablen Umschalten der Visualisierungen ebenso mittels Drop-Down Menü erhalten. Aus demselben Grund wurde ein Beibehalten der flexiblen, individuellen Achsenauswahl für den Parallelen Koordinaten Plot sowie den Scatterplot integriert.

Eine sinnvolle Erweiterung der Daten ist der Einbezug von (Nutzer-)Kritiken und deren Auswirkungen auf die Verkaufszahlen. Auch Analysen auf Korrelationen nicht öffentlich zugänglicher

Daten zu Entwicklungskosten und Umsätzen pro Spiel bieten Potenzial. Dies ist jedoch Teil der angesprochenen weiterführenden Marktstudien. Interessant ist eine weitere Unterteilung der Region *Rest of World*, die unter anderem Afrika, Asien und Südamerika enthält. Bezüglich der Visualisierungstechniken können Verbesserungen und weitere Kopplungen von Interaktionen direkt in den Plots Sinn ergeben, sind jedoch im Kontext der Anwendungsaufgaben eine zusätzliche Spielerei. Mehr Potenzial bieten Ansätze aus den *verwandten Arbeiten*. So ist ein automatisches Anordnen der Achsen je nach Korrelationsstärke im Parallelen Koordinaten Plots nach Hao et al. interessant.[7] Zur Verbesserung der Interaktionen auf der vielfältigen Wahl der Endgeräte ergibt eine Implementierung einer Komposition von Visualisierungen durch Zusammenschieben nach Chakraborty und Stuerzlinger Sinn.[2]

Insgesamt konnten die Anforderungen an die Visualisierungen zur Lösung des Zielproblems zufriedenstellend umgesetzt werden.

Anhang: Git-Historie

Literatur

- [1] Jacques Bertin und Wolfgang Scharfe. *Graphische Darstellungen und die graphische Weiterverarbeitung der Information*. Berlin: De Gruyter, 1982. ISBN: 978-3-11-006900-6. URL: <http://www.degruyter.com/doi/book/10.1515/9783110871494>.
- [2] Supratim Chakraborty und Wolfgang Stuerzlinger. “VizInteract: Rapid Data Exploration Through Multi-touch Interaction with Multi-dimensional Visualizations”. In: *Human-Computer Interaction - INTERACT 2021*. Hrsg. von Carmelo Ardito u. a. LNCS sublibrary, SL 3, Information systems and applications, incl. internet/web, and HCI. Cham, Switzerland: Springer, 2021, S. 610–632. ISBN: 978-3-030-85613-7.
- [3] Winnie Wing-Yi Chan. *A Survey on Multivariate Data Visualization*. 2006. URL: <https://people.stat.sc.edu/hansont/stat730/multivis-report-winnie.pdf> (besucht am 03.12.2022).
- [4] Evan Czaplicki, Hrsg. *Maybe*. o.J. URL: <https://package.elm-lang.org/packages/elm/core/latest/Maybe> (besucht am 10.10.2022).
- [5] Jürgen Fleig. *Marktanalyse und Marktforschung: Definition, Zweck und Beispiele*. 2020. URL: <https://www.business-wissen.de/hb/marktanalyse-und-marktforschung-definition-zweck-und-beispiele/> (besucht am 25.11.2022).
- [6] Michael Friendly und Daniel Denis. “The early origins and development of the scatterplot”. In: *Journal of the history of the behavioral sciences* 41.2 (2005), S. 103–130. ISSN: 0022-5061. DOI: 10.1002/jhbs.20078.
- [7] Ming C. Hao u. a. “Intelligent Visual Analytics Queries”. In: *IEEE Symposium on Visual Analytics Science and Technology, 2007*. Hrsg. von William Ribarsky. Piscataway, NJ: IEEE Service Center, 2007, S. 91–98. ISBN: 978-1-4244-1659-2.
- [8] Alfred Inselberg und Bernard Dimsdale. “Parallel Coordinates for Visualizing Multi-Dimensional Geometry”. In: *Computer Graphics 1987*. Hrsg. von Tosiya L. Kunii. Tokyo: Springer Japan, 1987, S. 25–44. ISBN: 978-4-431-68059-8.
- [9] John Lamping, Ramana Rao und Peter Pirolli. “A focus+context technique based on hyperbolic geometry for visualizing large hierarchies”. In: *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '95*. Hrsg. von Irvin R. Katz u. a. New York, New York, USA: ACM Press, 1995, S. 401–408. ISBN: 0201847051. DOI: 10.1145/223904.223956.
- [10] Dirk J. Lehmann u. a. “Visualisierung und Analyse multidimensionaler Datensätze”. In: *Informatik-Spektrum* 33.6 (2010), S. 589–600. ISSN: 0170-6012. DOI: 10.1007/s00287-010-0481-z.

- [11] o.A. *Videospiele: Weltweit*. 2022. URL: <https://de.statista.com/outlook/dmo/digitale-medien/videospiele/weltweit#umsatz> (besucht am 25.11.2022).
- [12] o.A. *XBox Series X/S Jubiläum: Die Bilanz nach dem ersten Jahr*. 2021. URL: <https://www.gameswirtschaft.de/wirtschaft/xbox-series-x-s-geburtstag-analyse/> (besucht am 25.11.2022).
- [13] H-J Schulz, S. Hadlak und H. Schumann. “The Design Space of Implicit Hierarchy Visualization: A Survey”. In: *IEEE transactions on visualization and computer graphics* 17.4 (2011), S. 393–411. DOI: 10.1109/TVCG.2010.79.
- [14] Heidrun Schumann und Wolfgang Müller. *Visualisierung*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000. ISBN: 978-3-540-64944-1. DOI: 10.1007/978-3-642-57193-0.
- [15] SID_TWR. *Video Games Sales Dataset: Video Games Sales & Game Ratings Data Scraped from VzCharts*. URL: https://www.kaggle.com/datasets/sidtwr/videogames-sales-dataset?select=XboxOne_GameSales.csv (besucht am 13.11.2022).
- [16] F. Tenzer. *Monatliche Verkaufszahlen der Xbox One in Europa bis September 2022*. 2022. URL: <https://de.statista.com/statistik/daten/studie/311733/umfrage/absatz-der-xbox-one-pro-monat-in-europa/> (besucht am 25.11.2022).
- [17] John Q. Walker. “A node-positioning algorithm for general trees”. In: *Software: Practice and Experience* 20.7 (1990), S. 685–705. ISSN: 00380644. DOI: 10.1002/spe.4380200705.
- [18] Edward J. Wegman und Qiang Luo. “High Dimensional Clustering Using Parallel Coordinates and the Grand Tour”. In: *Classification and Knowledge Organization*. Hrsg. von Rüdiger Klar und Otto Opitz. Studies in Classification, Data Analysis, and Knowledge Organization. Berlin und Heidelberg: Springer, 1997, S. 93–101. ISBN: 978-3-642-59051-1.