

Inhaltsverzeichnis

1. Einführung	3
1.1. Python.....	3
1.2. Qlik	3
1.3. Datensatz.....	4
2. Vergleich Python und Qlik	5
2.1. Business Understanding.....	5
2.2. Data Understanding.....	5
2.3. Data Preparation.....	6
2.4. Modelling	8
2.4.1. Veränderung der globalen Selbstmordraten	8
2.4.2. Prozentualer Anteil von Flüchtlingen an der Weltbevölkerung	10
2.4.3. Lebenserwartung / Erwartete Bildungsjahre.....	10
2.4.4. Historische Entwicklung der Säuglingssterblichkeit und Geburtenrate	12
2.4.5. Korrelation des BIP pro Kopf mit World Happiness Score	14
2.4.6. Weitere Grafiken	15
2.5. Sonstige wichtige Vergleichskriterien	16
3. SWOT-Analyse.....	18
3.1. Python.....	18
3.1.1. Data Scientist	18
3.1.2. Betrieblicher Entscheider	19
3.2. Qlik	20
3.2.1. Data Scientist	20
3.2.2. Betrieblicher Entscheider	21
4. Fazit.....	22

1. Einführung

Wie bei vielen unternehmensbezogenen Prozessen, geht es darum mit möglichst geringem Aufwand bestmögliche Ergebnisse liefern zu können.

Genauso ist es auch bei Datenwissenschaftlern. Man muss wissen, welche Tools sich am besten für die jeweiligen Anwendungszwecke eignen, um Daten adäquat visualisieren zu können.

Dementsprechend soll diese Ausarbeitung eine SWOT-Analyse der beiden Tools Python und Qlik beinhalten, damit veranschaulicht werden kann, in welchen Bereichen diese beiden Nutzmittel ihre Vor- und Nachteile haben und wie sie genutzt werden können.

Somit sollen zukünftige Entscheidungen für die Wahl des Visualisierungstools erleichtert werden, der erkenntlich werden soll, für welchen Einsatz das jeweilige Tool besser geeignet ist.

1.1. Python

Python ist eine universelle und höhere Programmiersprache, die den Anspruch hat, einen gut lesbaren, knappen Programmierstil zu fördern.

Für die Data Visualization wird sie zudem sehr häufig verwendet, da sie mit ihren diversen Bibliotheken eine umfassende Grundlage für das Data Modelling darstellt.

Für den ausgewählten Datensatz soll nun im Folgenden und unter Berücksichtigung von mannigfaltigen Kriterien, analysiert werden, wie gut Visualisierungen mit Hilfe der Programmiersprache erstellt werden können.

1.2. Qlik

Qlik ist ein US-amerikanisches Softwareunternehmen, welches verschiedene Produkte bezüglich des Umgangs mit Daten anbietet.

Zudem umfasst es auch eine cloudbasierte End-to-End Analytics-Plattform für Echtzeit-Datenintegration und -analyse. Diese ist laut der Website von Qlik auf die Nutzung in Unternehmen ausgerichtet, bietet allerdings auch eine kostenlose Test-Version, welche auch von privaten Nutzern für 30 Tage getestet werden kann.

Laut Hersteller verspricht das Tool „eine Data Fabric für moderne Datenarchitekturen und Analysen der nächsten Generation“, wie auch „hochgradig, interaktive, kontextbezogene Dashboards mit blitzschnellen Steuerelementen.“

Jene Test-Version soll im weiteren Verlauf des Projekts, äquivalent zu Python, unter der Beachtung von ausgewählten Kriterien beleuchtet werden.

1.3. Datensatz

Der bereits erwähnte und bearbeitete Datensatz stammt von der Quelle „UN Data“ und befasst sich mit dem Thema Human Development, anhand welchem der spätere Vergleich zwischen Python und dem benutzten Tool, Qlik, durchgeführt werden soll.

Dies hat zum einen den Vorteil, dass die Online-Seiten der „United Nations“ als Quellen vertrauenswürdig sind und zum anderen von offiziellen Seiten die präsentierten Daten regelmäßig aktualisiert werden.

Die Daten werden hierbei jeweils in einzelnen kleinen Datensätzen präsentiert und zu jedem Land über einen Zeitraum von über 60 Jahren für einzelne Kategorien aufgeführt.

Additiv werden noch weitere Datensätze der WHO für den Toolvergleich herangezogen.

Dementsprechend kann nicht von einem großen Datensatz, sondern mehreren kleinen Datensätzen gesprochen werden, die im weiteren Verlauf der Analyse benutzt werden sollen.

Obwohl der Datensatz mehr oder weniger nur Mittel zum Zweck für die Tool-Analyse ist, so wurde dieser dennoch bewusst gewählt. Immerhin können mit dem Thema Human Development verschiedenste Fragen zum Leben der Menschen auf der Welt beantwortet werden, um basierend hierauf auch Entscheidungsfindungen in Politik und Wirtschaft in die Wege zu leiten. Somit betrifft dieses Thema indirekt die gesamte Gesellschaft und stellt in Kombination mit dem Vergleich von Qlik und Python eine interessante Aufgabe dar.



Die Mehrheit der zehn kleinen Datensätze wurde anschließend für die weitere Analyse zusammengeführt, um damit leichter arbeiten zu können. Lediglich die CSV-Dateien „WHR2021“, „natural_disasters“ und „suicide_rates“ werden aufgrund der zu hohen Differenz an Zeilen und Spalten weiterhin separat betrachtet.

2. Vergleich Python und Qlik

Um den Vergleich der beiden Tools objektiv durchführen zu können, wird hierfür ein Bewertungsbogen hinzugenommen, der vielfältige Kriterien für die Analyse bereithält. Diese orientieren sich am CRISP-DM Prozess und werden in die einzelnen Teilschritten von diesem unterteilt.

Des Weiteren sollen auch spezifische Visualisierungen zum Thema Human Development erstellt werden, wofür einzelne Fragen aufgestellt werden.

Hierfür sollen für jeweils beide Tools die Resultate möglicher Visualisierungen gezeigt werden und welche Fragen weniger adäquat mit Visualisierungen basierend auf dem Datensatz beantwortet werden können.

2.1. Business Understanding

Zu diesem ersten Schritt des CRISP-DM Prozesses können keine Kriterien angewandt werden, da das Business Understanding, in diesem Falle also das Thema Human Development, nicht in den einzelnen Tools durchgeführt wird. Hierfür kann eine Internet-Recherche benutzt werden, allerdings eignen sich verschiedene Visualisierungen nicht dafür, um das untersuchte Thema an sich zu erklären.

2.2. Data Understanding

Anders als das Business Understanding, kann das Data Understanding als erster Schritt mit Kriterien für die beiden Tools dienen. Hierbei soll vor allem untersucht werden, inwiefern es möglich ist, neue Daten in die Tools hinzuzufügen, übersichtlich anzuordnen und diese für erste Visualisierungen ohne weiteren Aufwand zu benutzen.

In Qlik funktioniert das Hinzufügen von einzelnen Datensätzen über ein sogenanntes Datenmanager-Feld, in welchem beispielsweise einzelne CSV-Dateien manuell ausgewählt und in den jeweiligen Ordner geladen werden können. Dieser Schritt ist durch eine gute Beschreibung der verschiedenen Features gut durchzuführen, selbst wenn man zuvor noch nie mit der Oberfläche gearbeitet hat.

Des Weiteren kann hierbei festgehalten werden, dass sich neue Daten auch in sehr geringer Zeit einfügen lassen. Bei den für die Analyse benutzen CSV-Dateien, sind in diesem Prozess nur wenige Sekunden verstrichen.

In der Hauptübersicht, wo auch die verschiedenen Visualisierungstypen zu finden sind, werden dann die einzelnen Kategorien der geladenen Datensätze in der Seitenleiste angezeigt, wodurch ein erster Überblick über die vorhandenen Daten möglich ist.

Solange nur ein Datensatz mit wenigen Kategorien geladen wurde, ist dies auch recht übersichtlich. Jedoch muss bei größeren Datensätzen mit mehreren Spalten etwas genauer gesucht werden, um spezifische hiervon identifizieren zu können. Wurden allerdings mehrere Datensätze in den gleichen Ordner geladen, kann auch zwischen einzelnen Datensätzen gewählt werden, welche an der Seitenleiste angezeigt werden sollen, was zusätzliche Übersicht garantiert.

Insofern die geladenen Datensätze dann schon nutzbare Daten enthalten, können im Folgenden auch ohne geringen Aufwand erste Testvisualisierungen für die initiale Sichtung und Einschätzung über die Qualität der Daten getätigt werden.

In dieser Hinsicht hat Qlik in der Analyse mehrere Kriterien durchaus erfüllt.

Gegenüberstellend sollen die zu Beginn genannten Kriterien nun auch auf die Programmiersprache Python angewandt werden. Hierfür wird allerdings immer vorausgesetzt, dass die nötigen Programmierkenntnisse vorhanden sind.

Unter Berücksichtigung dessen funktioniert auch hier das Einfügen von neuen Files mit Hilfe der Bibliothek „pandas“ relativ einfach, wofür auch keine richtigen Datengrößen als Richtlinie dienen können, ab wann das Laden von Datensätzen nicht mehr möglich ist. Dies funktioniert äquivalent zu Qlik ohne übermäßige Ladezeiten. Hierbei beansprucht jedoch das Schreiben des notwendigen Codes für diesen Schritt einige Minuten mehr als das Auswählen von Dateien für den Datenmanager in Qlik.

Weiter ist in Python auch eine initiale Datenexploration durch die `head()`- und `info()`-Funktionen möglich, um einen schnellen Überblick zu erhalten und Null-Werte anzeigen zu lassen. Allerdings weist die Übersichtlichkeit der geladenen Daten hierbei Defizite auf, da vor allem bei umfangreicheren Datensätzen das Auffinden spezifischer Zeilen oder Spalten per bloßem Auge nicht direkt gewährleistet ist.

Nichtsdestotrotz bietet Python einen anderen Vorteil, der so nicht direkt in Qlik existiert: das Laden durch eine API, sodass nicht einzelne Files manuell hinzugefügt werden müssen.

Damit erfüllt auch Python einige der obigen Bewertungsmaßstäbe. Zwar erbringen das Schreiben des Codes für das Laden und die etwas fehlende Übersichtlichkeit Nachteile gegenüber Qlik, werden aber durch den möglichen Einsatz von APIs ausgeglichen.

2.3. Data Preparation

Im dritten Schritt des CRISP-DM Prozesses wird für gewöhnlich Bezug auf die Präparation der Daten genommen, sodass im weiteren Arbeiten mit dem jeweiligen Datensatz keine Probleme entstehen. Dabei muss vor allem Wert auf das Data Cleaning gelegt werden, ob und inwiefern Inkonsistenzen, Null-Werte, einzelne Zeilen oder Spalten entfernt werden können.

Aber auch das Prüfen, ob Datenanreicherungen, Strukturänderungen und Datentypänderungen am Datensatz vorgenommen werden können, darf hierbei nicht zu kurz geraten.

In Python ist das erwähnte Data Cleaning zum Beispiel recht flexibel. Ist die nötige Erfahrung im Umgang mit der Sprache vorhanden, können unterschiedliche Befehle benutzt werden, um gewünschte Werte zu entfernen.

So konnten die gewählten Datensätze zum Thema Human Development sehr schnell von Null-Werten und anderen unbrauchbaren Inhalten gereinigt werden, um zeitnah erste Visualisierungen erstellen zu können. Auch die Umstrukturierung der einzelnen Datenmengen war ohne Probleme möglich. Hierfür erwiesen sich primär Online-Foren und ChatGPT als sehr hilfreich und brauchbar.

Außerdem konnten zusätzlich gewünschte Kriterien als Spalten sehr gut zu den einzelnen Datensätzen hinzugefügt werden, um somit ein größeres Kategorienspektrum für die spätere Auswertung zu ermöglichen und auch Datentypänderungen mit Python ohne Probleme ausgeführt werden.

Hierbei wurden spezifisch ausgewählte Regionen aus dem Datensatz entfernt wie zum Beispiel Zentralasien, Mittelfrika und Europa, sodass am Ende nur noch die einzelnen Länder zum Vergleichen übrig bleiben.

Im Gegensatz hierzu erwies sich Qlik in diesem Schritt als weniger brauchbar, da innerhalb des Tools ein wirkliches „Cleanen“ von importierten Datensätzen gar nicht erst möglich war. Zwar konnten im Prozess des Visualisierens einige Kategorien oder Werte gefiltert, aber nicht gänzlich aus den Datensätzen gelöscht, werden.

Weiter waren auch Änderungen an Datentypen oder gar der Struktur der einzelnen Datensätze nicht durchzuführen, sodass die importierten Daten zuvor gänzlich mit Python gecleaned werden mussten, da dies im Tool nicht mehr möglich war.

Auch das Zusammenfügen einzelner Datenmengen erwies sich als eher unpraktisch. Zwar konnten diese im sogenannten Datenmanager verknüpft werden, allerdings traten dann Probleme bei der Modellierung auf, da die Spalten und Zeilen nicht genau übereinstimmten und wie bereits erwähnt, das Ändern dieser im Tool selbst nicht möglich ist.

Dementsprechend kann für diesen Prozessschritt festgehalten werden, dass Python für die Anpassung von Daten für die weitere Benutzung definitiv die bessere Alternative gegenüber Qlik darstellt.

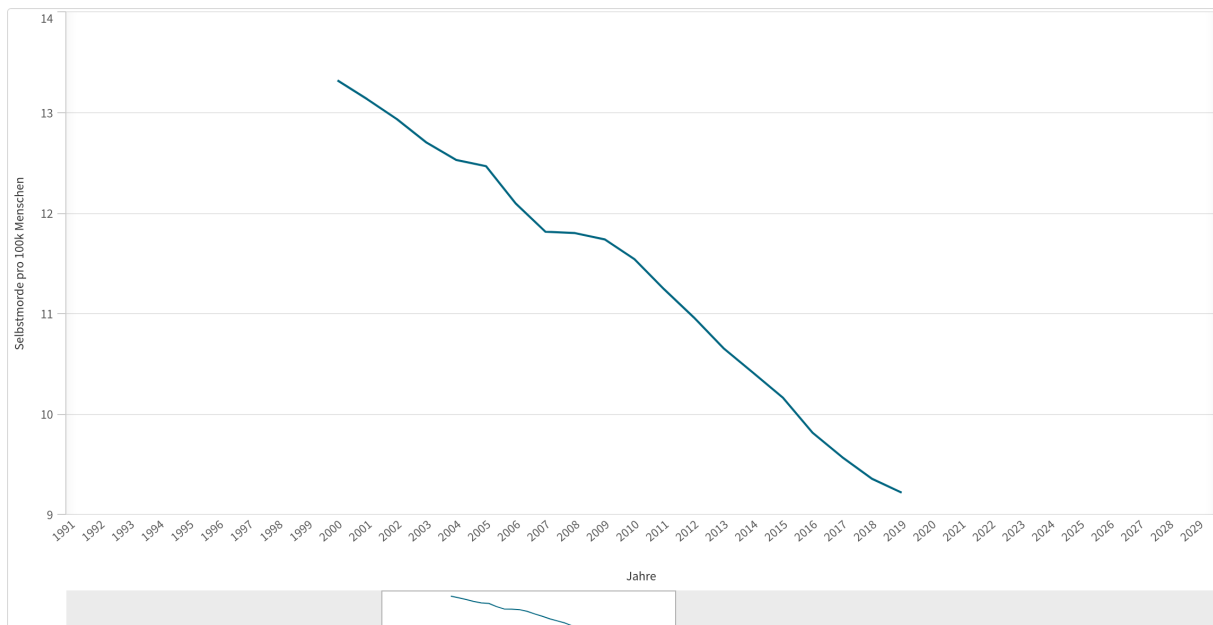
2.4. Modelling

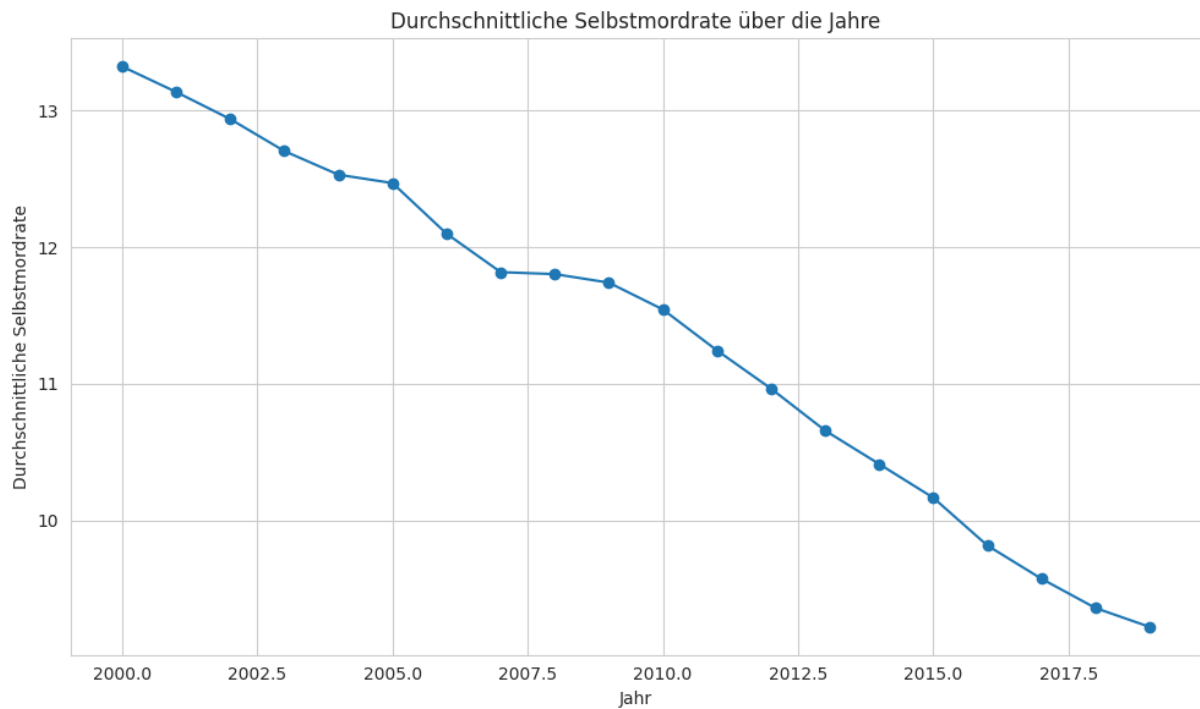
Nachdem in Python alle Daten für die Weiterverarbeitung vorbereitet wurden, kann mit den ersten Visualisierungen begonnen werden.

Dafür sollen verschiedene Fragen bezüglich Human Development aufgestellt werden, um zu testen, inwiefern diese mit den gegebenen Modellierungsmöglichkeiten in Python oder Qlik beantwortet werden können.

Diese sollen im Folgenden gegenübergestellt und anhand dessen die einzelnen Vor- sowie Nachteile aufgezeigt werden. Die erste Grafik stellt hierbei immer die Visualisierung dar, welche mit Qlik erstellt wurde. Die Letztere ist immer von Python.

2.4.1. Veränderung der globalen Selbstmordraten





In diesem Falle werden die durchschnittlichen Selbstmordraten im Zeitraum von 2000 bis 2019 visualisiert.

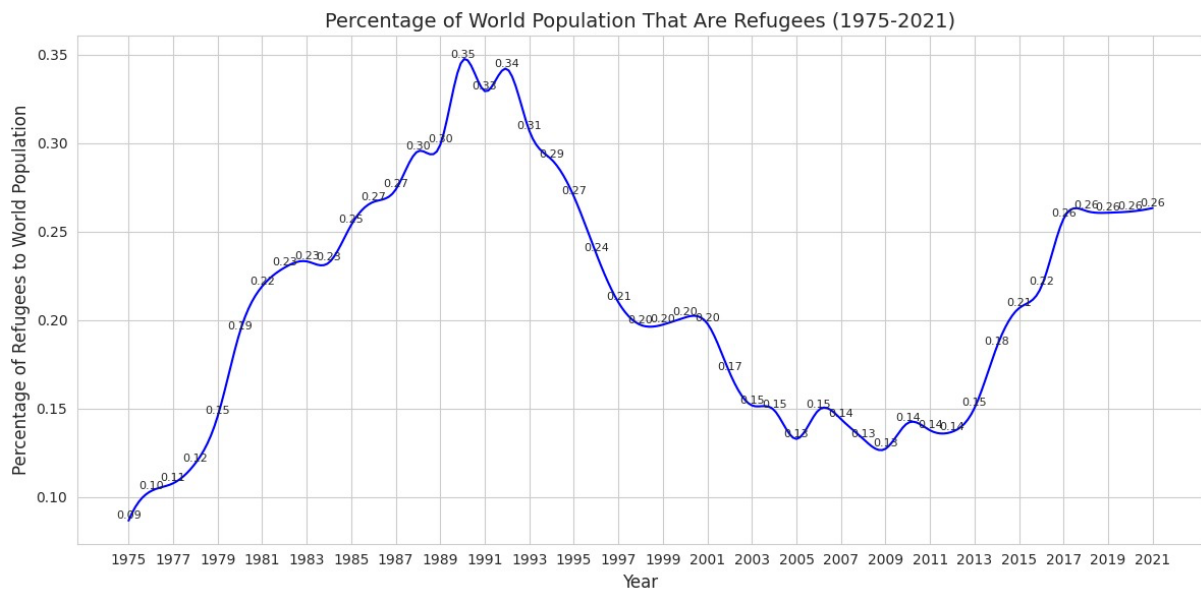
Tatsächlich können solchen Arten von Visualisierungen in Qlik anhand der Angabe von zwei verschiedenen Dimensionen recht einfach und auch intuitiv erstellt werden.

Auch die Qualität dieser ist sichtbar zu validieren, was aber auch für Python gilt. In Python hängt dies meistens von der verwendeten Bibliothek ab, ist aber nur in den wenigsten Fällen kritisierbar.

Allerdings ist hierbei das Erstellen nicht so intuitiv wie in Qlik möglich. Hierfür ist erstmal das Schreiben des Codes notwendig, weshalb mit den fertigen Datensätzen das Modellieren von zum Beispiel Linien- oder Balkendiagrammen deutlich länger dauert als in Qlik, in welchem man einfach nur die Art des Modells auswählt und Dimensionen hinzufügt.

Dies ist ein großer Vorteil gegenüber Python, da auch ohne vorherige Nutzung des Tools schnell Ergebnisse geliefert werden können.

2.4.2. Prozentualer Anteil von Flüchtlingen an der Weltbevölkerung



Weiter sollte dann auch untersucht werden, wie sich die Flüchtlingszahlen im Vergleich zur Weltbevölkerung über die Jahre entwickelt haben.

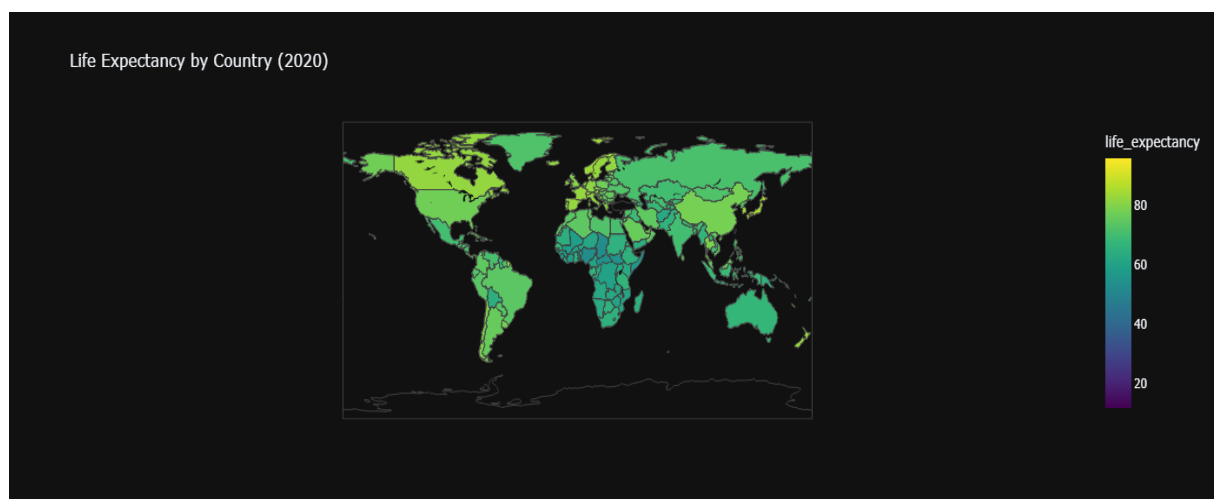
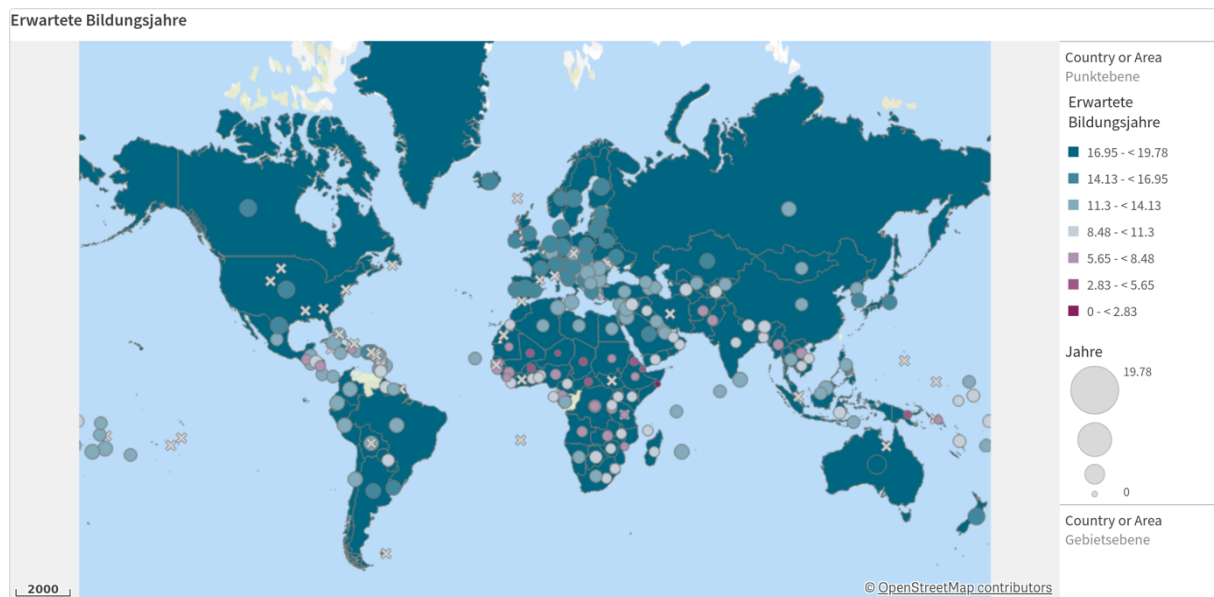
Auch hierfür reichte ein einfaches Liniendiagramm aus, um Erkenntnisse über die Veränderung der letzten Jahrzehnte zu erhalten.

Die Schwierigkeit lag hierbei aber eher daran, dass basierend auf dem Datensatz die prozentualen Werte errechnet werden mussten. Da der fertig zusammengefügte Datensatz nur eine Spalte mit „Refugee Origin“ und absoluten Zahlen enthielt, mussten diese erst durch die Kategorie „Total Population“ geteilt werden, um das obige Ergebnis zu erhalten.

Dementsprechend kann in diesem Fall nur die Visualisierung in Python aufgezeigt werden, da in diesem Fall in Qlik die Rechnung äquivalent durchzuführen gewesen wäre, was aber nicht möglich war. Wie im Abschnitt „Data Preparation“ bereits beschrieben, hätte hierfür innerhalb des Tools eine neue Spalte für die prozentualen Werte erstellt werden müssen. Dies war allerdings nicht möglich, sodass diese Art von Grafik nur mit Hilfe von Python und nicht mit Qlik erstellt werden konnte.

2.4.3. Lebenserwartung / Erwartete Bildungsjahre

Um im Laufe der Analyse auch zu testen, wie die Funktionalität der beiden Tools mit Geo-Daten ist, sollten auch verschiedene Maps erzeugt werden, um hierbei den Umgang damit und auch die Individualisierbarkeit aufzuzeigen.



Hierfür wurden zum einen in Qlik eine Weltkarte für die erwarteten Bildungsjahre und in Python eine Weltkarte für die Lebenserwartung pro Land für das Jahr 2020 erstellt.

Wie man sehen kann, zeugen auch diese beiden Grafiken von einer sehr guten Qualität und geben Aufschlüsse für die Daten der einzelnen Länder.

Weiter ist es auch in beiden Tools problemlos möglich gewisse Einstellungen an der Visualisierung anzupassen. So können beispielsweise in Qlik die Größen der Kreise und auch die Farbskalierung ohne erheblichen Aufwand verändert werden. Ebenfalls kann auch in Python die Farbskala beliebig angepasst werden.

Vor allem in Qlik können auch sehr benutzerfreundlich, wie zum Beispiel in MS PowerPoint, die Namen der Grafiken und Achsen in Textfeldern geändert werden.

Ferner wurde auch auf das Kriterien der Geschwindigkeit geachtet, in welcher die beiden Modelle erstellt wurden. Hierfür ist der manuelle Aufwand in beiden Fällen ungefähr ähnlich. In Python nimmt vor allem das Schreiben des Codes etwas mehr Zeit in Anspruch. Aber auch in Qlik muss bei erstmaliger Nutzung erstmal genauer betrachtet werden, wo die einzelnen

Dimensionen, Ebenen und Skalierungen ausgewählt werden können, sodass auch hier etwas mehr Zeit eingeplant werden musste als zum Beispiel bei dem vorherigen Liniendiagramm. Glücklicherweise findet man aber sowohl zu Python als auch zu Qlik gute Erklärungen und Dokumentationen im Internet, wodurch dieser Prozess etwas erleichtert werden kann. Sind aber die grundlegenden Einstellungen für die Diagramme fertig, werden diese, mit wenigen Ausnahmen, schnell und ohne Zeitverzug angezeigt. Natürlich gibt es abgesehen von Python auch andere Programmiersprachen, die womöglich etwas schneller, aber nicht für die Data Visualization zu gebrauchen sind. Somit kann festgehalten werden, dass bei adäquaten Datensätzen beide Tools einen guten Umgang mit Geo-Daten offenbaren und sehr gut zu benutzen sind.

2.4.4. Historische Entwicklung der Säuglingssterblichkeit und Geburtenrate

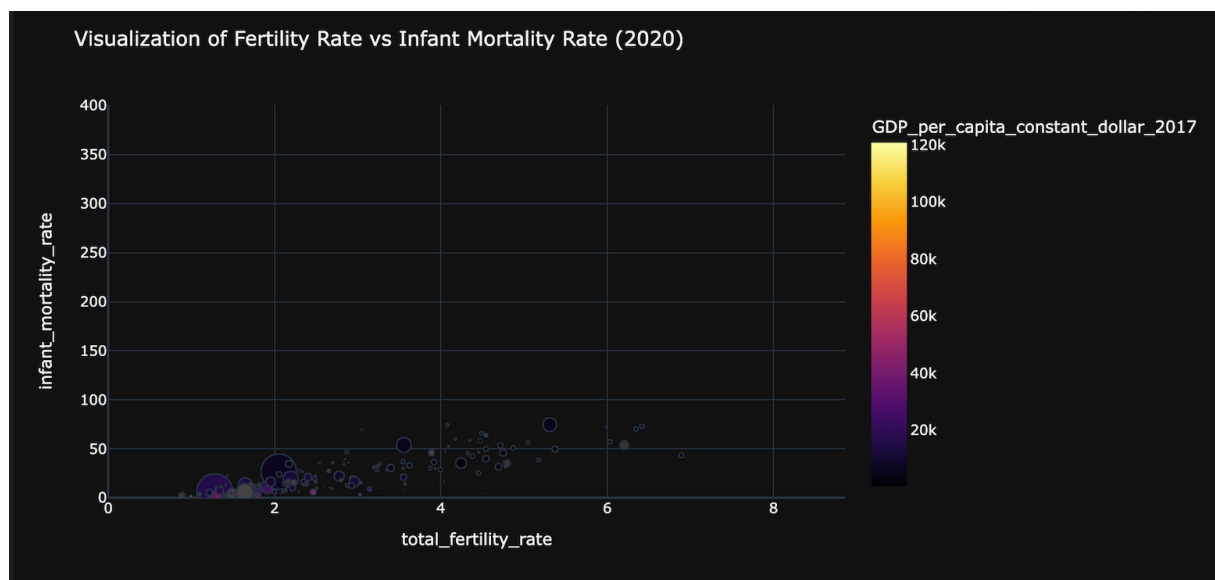
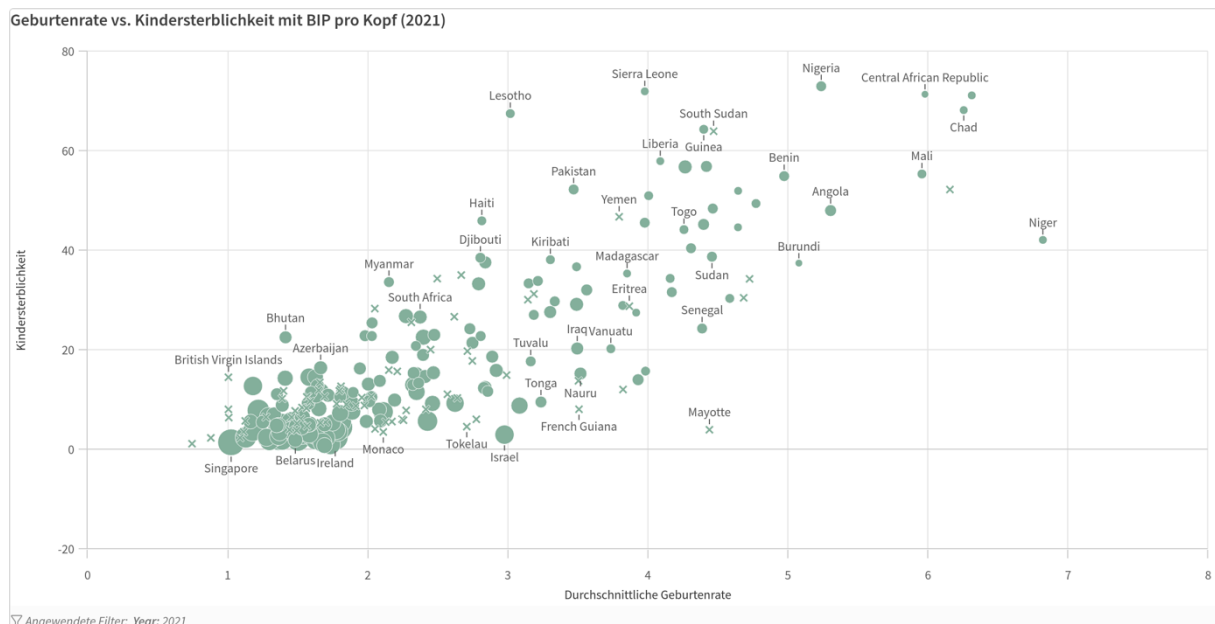
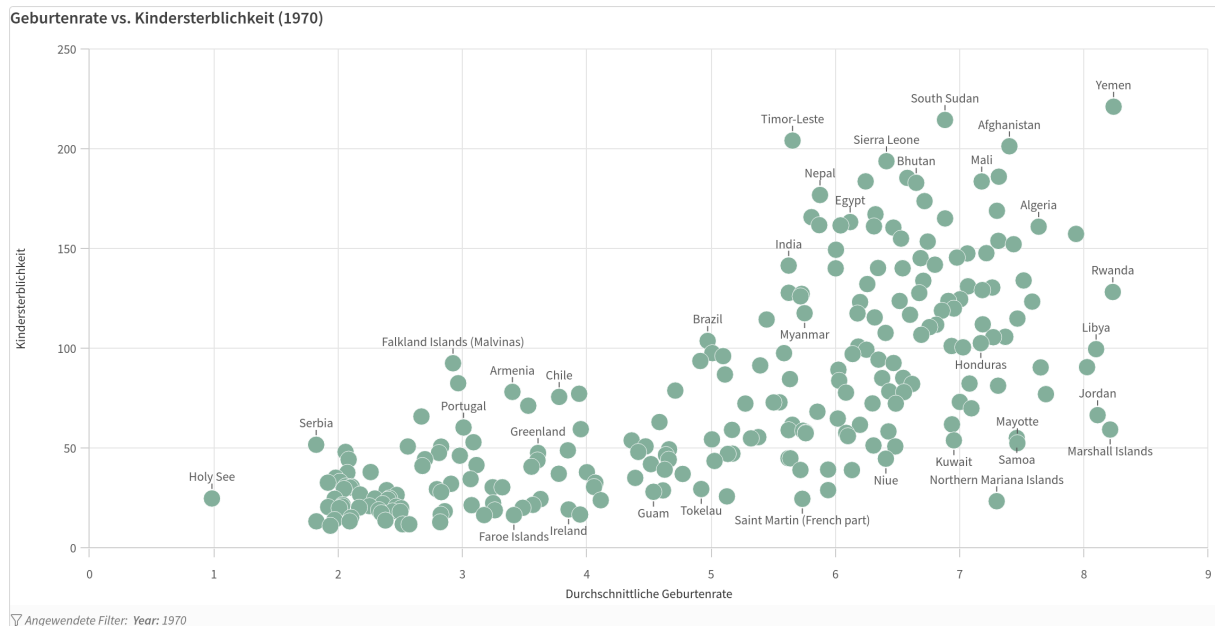
Nachdem die Nutzbarkeit der beiden Tools im Umgang mit Geo-Daten analysiert wurde, sollen noch einige Scatter-Plots erstellt werden. In diesem Fall soll die Korrelation zwischen der Säuglingssterblichkeit mit der Geburtenrate der einzelnen Länder angezeigt werden. Hierfür erwies sich vor allem Qlik als sehr leicht zu handhaben und übersichtlich. Für die Erstellung der folgenden Grafik bedurfte es nämlich nur der Auswahl des Diagrammtyps und der beiden Dimensionen aus dem Datensatz.

Auch das Hinzufügen einer dritten Dimension durch das BIP pro Kopf, welches die Kreisgröße regelt, stellte hierbei keine Herausforderung dar (zweite Grafik, Seite 11).

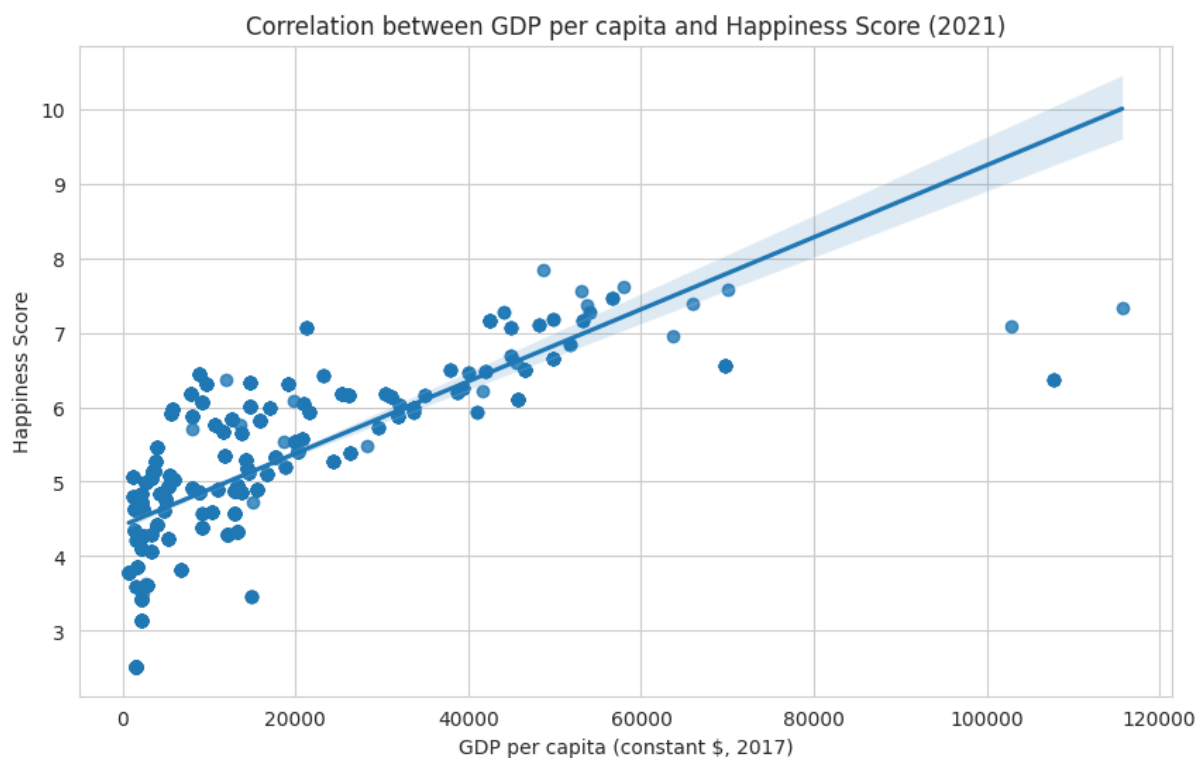
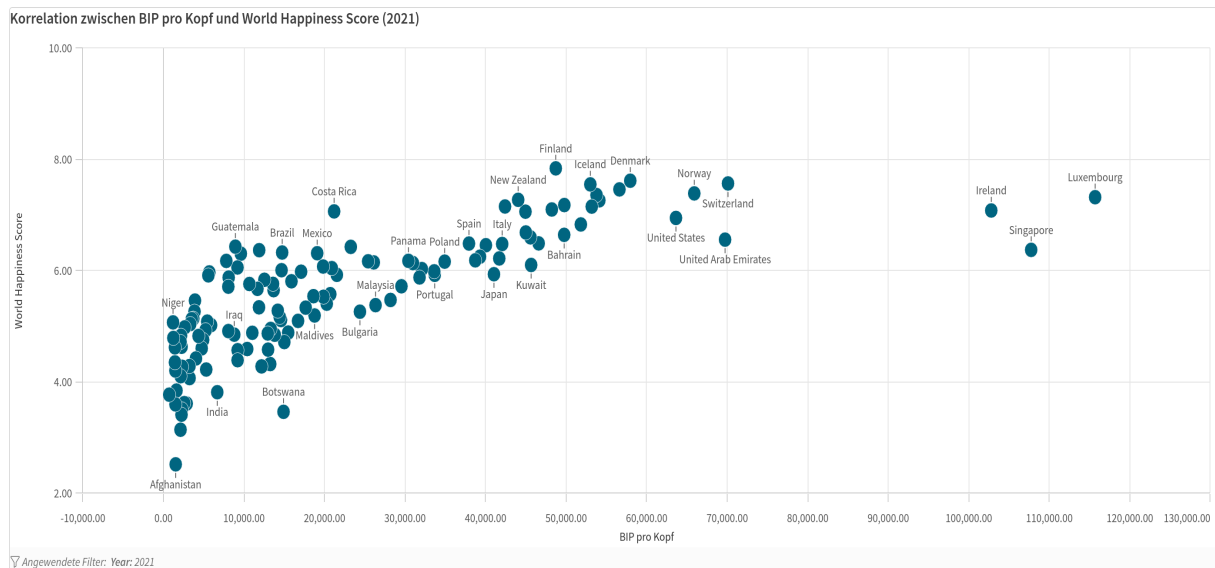
Lediglich jene Länder, für die kein BIP pro Kopf im Datensatz vorhanden war, werden als Kreuz visualisiert.

Im Gegensatz hierzu, war der Aufwand in Python, wie auch im Abschnitt der Selbstmordraten erwähnt, deutlich höher, da nicht einfach eine Visualisierungsmethode und die jeweiligen Kategorien ausgewählt werden konnten.

Auf Dauer stellt sich heraus, dass vor allem leichtere Grafiken wie Balken-, Linien- oder auch Punktdiagramme deutlich aufwendiger in Python sind als in Qlik.



2.4.5. Korrelation des BIP pro Kopf mit World Happiness Score



Additiv zu den vorherigen Scatter-Plots, wurde auch versucht eine Regressionslinie bezüglich der Korrelation von World Happiness Score und dem BIP pro Kopf einzufügen.

Tatsächlich hat dies in Qlik in diesem Visualisierungsversuch nicht funktioniert. Auch mit Nachforschen im Internet und in Dokumentationen erschien es weiterhin unklar, da man bei anderen Modellierungstypen durchaus diese Linie als Zusatz einfügen konnte.

Dementsprechend konnte diese Art der Veranschaulichung nur in Python vollends beendet werden.

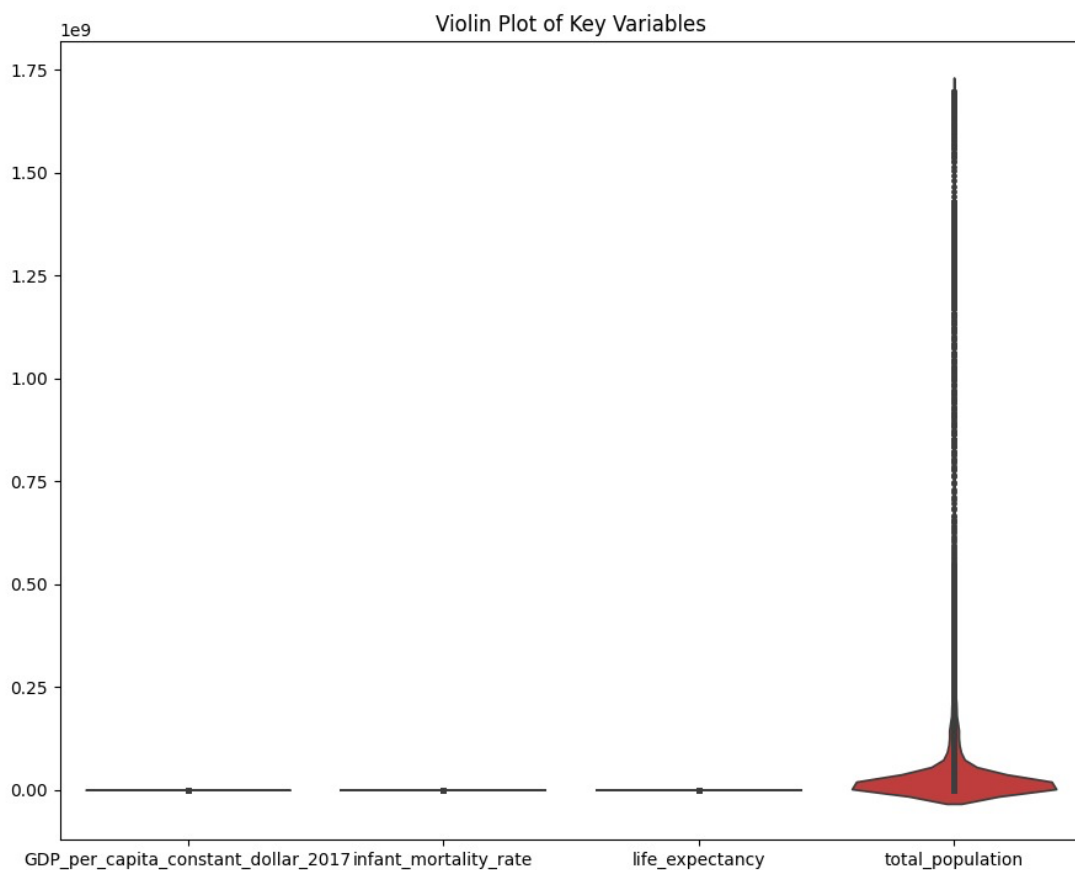
In diesem Falle erfüllt Qlik deshalb das Kriterium der Benutzerfreundlichkeit und Übersicht nicht ganz.

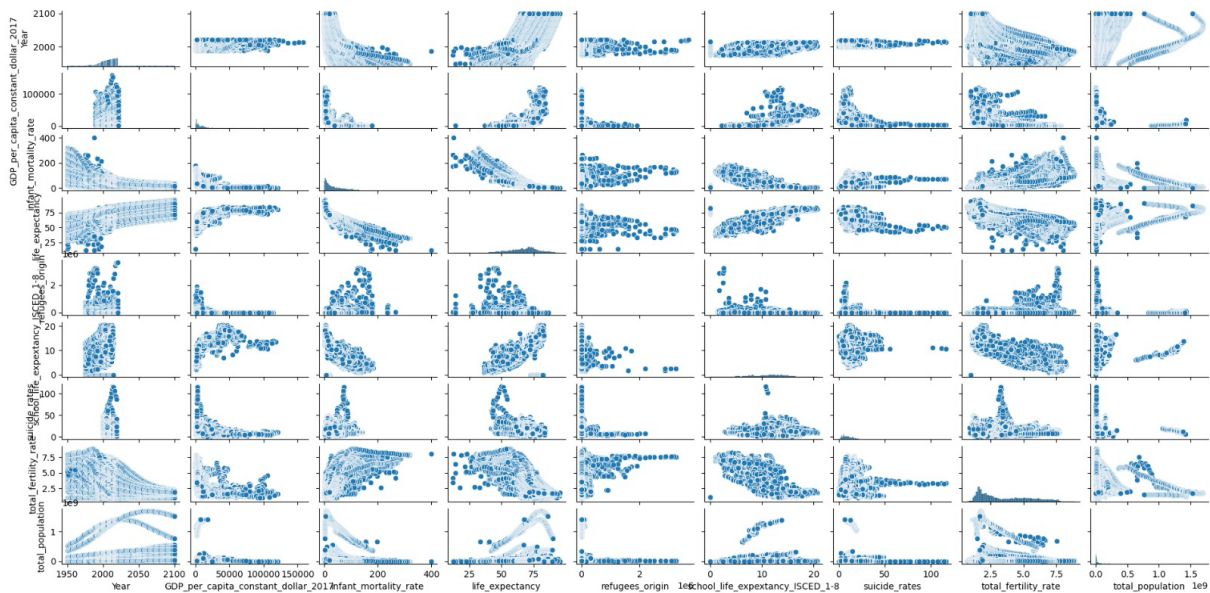
2.4.6.Weitere Grafiken

In der Vorlesung Data Visualization wurden zudem noch weitere Grafiken vorgestellt, welche mit den entsprechenden Datensätzen erzeugt werden können.

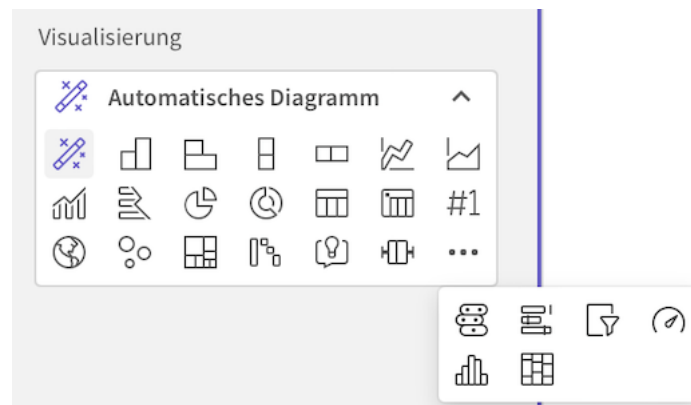
Dementsprechend sollen diese im Folgenden auch noch präsentiert werden. Zwar erfüllt die Aussagekraft in Bezug auf Human Development nicht ganz die gewünschten Anforderungen, vielmehr ging es hier aber darum zu zeigen, wie groß die Auswahlmöglichkeiten sind.

So wurden hierfür jetzt beispielhaft der Violin- und der Pair-Plot ausgewählt.





Diese beiden Typen von Visualisierungen gibt es in Qlik nicht, so dass sich feststellen lässt, dass Python eine deutlich größere Auswahl an Visualisierungsmöglichkeiten bereithält. Qlik verfügt zwar auch über diverse Modellierungstypen, ist aber im Vergleich zu Python eher eingeschränkt. So findet man additive Visualisierungen, wie Korrelationsmatrizen, nicht. Will man also Modellierung über die Standardgrafiken hinaus durchführen, besitzt Python gegenüber Qlik, gemessen an der Quantität der Auswahl, einen Vorteil. Der folgende Ausschnitt zeigt alle Visualisierungstypen, die das Tool für den Nutzer bereithält. Bei Python gibt es, je nach Einsatz von Bibliotheken, kaum eine Begrenzung.



2.5. Sonstige wichtige Vergleichskriterien

Bislang wurde der Toolvergleich primär am CRISP-DM Prozess durchgeführt. Allerdings gibt es auch mannigfaltige Parameter, die außerhalb davon erfüllt werden müssen, um ein möglichst effizientes Arbeiten innerhalb dieses Prozesses zu gewährleisten. Dazu zählt beispielhaft eine gute Dokumentation oder gute Erklärungen im Internet, wie bereits erwähnt wurde.

Auch muss die Stabilität des benutzten Tools gewährleistet sein. Immerhin darf dies nicht gleich abstürzen oder sich aufhängen, sobald größere Datenmengen ins Spiel kommen. Hierfür wurden aber im Laufe der Analyse keine größeren Anfälligkeiten festgestellt. Zwar merkt man in Python bei Visualisierungen wie dem Pair-Plot, dass das Rendern etwas länger dauert, aber das Tool trotzdem einwandfrei weiterläuft. Ebenfalls wurden auch bei Qlik in Bezug auf die importierten Datensätze keine Schwachstellen festgestellt.

Weiter ist es wichtig, dass die erstellten Visualisierungen exportiert werden können. Dies ist in Python mit fast allem möglich. Ebenso unterstützt dies auch Qlik, wobei hier sogar zwischen den drei Optionen, „Daten“, „Bild“ und „PDF“ gewählt werden kann und damit dieses Kriterium auch durchaus erfüllt.

Darüber hinaus werden heutzutage Aspekte wie kollaboratives Arbeiten immer wichtiger. Im Unternehmen will man nicht mehr darauf warten bis ein Kollege mit einem Dokument fertig ist, sondern gleichzeitig daran arbeiten.

Leider wurde diese Funktion in Qlik nicht wirklich ersichtlich. Zwar bietet Qlik laut Website dies für ein anderes ihrer Produkte „Qlik Talend“ an, jedoch nicht für das in dieser Analyse im Vordergrund stehende Datenvisualisierungstool.

Auf der Gegenseite ist Python ein Stück weiter, indem mit GitHub und anderen Möglichkeiten bereits Plattformen für gemeinsames Arbeiten unterstützt werden. Zwar ist dies am Anfang etwas kompliziert, funktioniert bei längerer Benutzung aber problemlos.

Ebenfalls muss die Kompatibilität mit anderen Plattformen geschaffen sein. Dafür müssen in Python zwar Schnittstellen vorhanden sein oder geladen werden, aber damit wird auch dieser Parameter erfüllt. In Qlik gab es bezüglich dessen während der Analyse auch keine Probleme, weshalb das Tool jenen Parameter wie Python genauso einhält.

Als letztes wichtiges Kriterium muss noch der finanzielle Aspekt beleuchtet werden.

Immerhin spielt es für den Endnutzer des Tools eine entscheidende Rolle, ob er sich dies überhaupt leisten kann. Ist die Finanzierungsmöglichkeit nämlich nicht gegeben, so können die meisten Kriterien schon außer Acht gelassen werden, da der Kunde durch den teuren Preis direkt abgeschreckt wird.

Hierbei hat Qlik in der Tat einen großen Nachteil gegenüber Python. Es gibt zwar eine kostenlose Test-Version, die der Anbieter für 30 Tage freischalten lässt und anhand jener auch die Analyse durchgeführt wurde, allerdings wird danach eine monatliche Gebühr von 30 US-Dollar fällig, um weiterhin die Dienste des Tools nutzen zu können. Somit muss sich der private Endnutzer, aber auch der mögliche Geschäftskunde, hinterfragen, ob es ihm das wert ist diesen Preis zu bezahlen, wenn es günstigere Alternativen gibt.

So beispielsweise Python. In Anbetracht dessen, dass die Grundversion und die meisten Bibliotheken kostenfrei zu erwerben sind und für jeden zugänglich sind, kann durchaus validiert werden, dass Python in dieser Hinsicht besser dasteht als Qlik.

Am Ende obliegt es aber jedem selbst, ob er den Preis dafür zahlen möchte.

3. SWOT-Analyse

Die SWOT-Analyse ist ein Instrument der strategischen Planung. Hierbei werden die vier Parameter „Strengths“, „Opportunities“, „Weaknesses“ und „Threats“ in einer Korrelationsmatrix miteinander in Verbindung gesetzt.

Da die späteren Endanwender zwischen einem Data Scientist und einem geschäftlichen Entscheider variieren können, sollen hierfür zwei angepasste Analysen aufgezeigt werden.

3.1. Python

3.1.1. Data Scientist

Als Data Scientist erhält man mit Python ein Tool, das sehr flexibel im Umgang mit Daten ist. Zudem ist es kostenlos und erhält diverse Möglichkeiten zur Visualisierung von Daten, womit die einzelnen Schritte des CRISP-DM Prozesses problemlos durchlaufen werden können.

Weiter bieten die umfangreichen Bibliotheken und ausführlichen Dokumentationen viele Optionen zur Erweiterung und Implementierung neuer Features. Somit ist Python vielseitig einsetzbar und dient auch mit dem Einsatz von künstlicher Intelligenz als Universalwerkzeug.

Außerdem bietet der Einsatz der Programmiersprache den Vorteil, dass bereits erlernte Kenntnisse nicht verlernt werden.

Nichtsdestotrotz gibt es immer auch irgendwelche Nachteile oder mögliche Gefahren. So ist Python, wie im vorherigen Toolvergleich bereits erwähnt, nicht immer sehr übersichtlich und erfordert auch gute Programmierkenntnisse, damit adäquate Ergebnisse erzeugt werden können. Besitzt man diese nicht, wird es schwer geeignete Visualisierung zu entwerfen.

Weiter sind im Umgang mit dem Datensatz zwar keine größeren Leistungsprobleme aufgetreten, dennoch gibt es andere Programmiersprachen, die etwas schneller arbeiten. Erahnen lassen konnte sich das anhand der Erstellung des Pair-Plots, da hierbei schon ein wenig mehr Zeit verstrichen ist, bis die Visualisierung sichtbar wurde.

Auch ist der zu betreibende Aufwand für gute Modellierungen durchaus als Schwäche anzuerkennen, was natürlich auch die Gefahr bringt, dass mit der Zeit Konkurrenz durch spezialisierte Software auf dem Markt entsteht, in welcher dieser Aufwand verringert wird.

Strengths <ul style="list-style-type: none"> • Kostenlos • Viele Visualisierungen möglich • Viele Bibliotheken • Durchführbarkeit CRISP-DM 	Opportunities <ul style="list-style-type: none"> • Als Universalwerkzeug nutzbar • Möglichkeiten zum Einsatz von KI
Weaknesses <ul style="list-style-type: none"> • Viel Aufwand • Programmierkenntnisse nötig • Nicht übersichtlich 	Threats <ul style="list-style-type: none"> • Konkurrenz durch spezialisierte Software • Leistungsprobleme bei großen Datenmengen

3.1.2. Betrieblicher Entscheider

Als betrieblicher Entscheider hat man differenzierte Präferenzen als ein Data Scientist, der jeden Tag mit diversen Daten arbeitet. So muss man auch Geschäftsdaten in einem Dashboard präsentieren, um später damit mögliche Geschäftsentscheidungen begründen zu können. Dementsprechend ist es hierfür notwendig, eine zweite SWOT-Analyse zu konfigurieren.

Für betriebliche Entscheider, für die nicht die besten Erfahrungen im Umgang mit Python zu erwarten sind, bietet die Programmiersprache vor allem Stärken, indem sie sehr umfangreiche Bibliotheken und auch viele unterstützende Dokumentationen besitzt. Damit kann vielen Unerfahrenen geholfen werden, geeignete Modellierungen zu konzipieren. Dies eröffnet zudem die Möglichkeit, spätere Dashboards in Hinsicht auf jede Komponente individuell anzupassen, weil die Bibliotheken nahezu unbegrenzt sind und gut erklärt werden.

Ferner kann Python auch in andere unternehmensinterne Software integriert werden, sodass es auch abseits der Data Visualization vielfach zum Einsatz kommen kann.

Allerdings stehen auf der gegenüberliegenden Seite auch einige Schwachstellen, welche Python in Bezug auf die Nutzung durch einen betrieblichen Entscheider offenbart. So gibt es zum Beispiel keine Standardisierung von Dashboards innerhalb des Tools. Vor allem bei mangelhafteren Programmierkenntnissen, kann es in diesem Fall schwer werden geeignete und anschauliche Dashboards für die Präsentation gegenüber Dritten zu erstellen. Diese fehlende Ästhetik birgt auch die Gefahr, dass dadurch eine gewisse Überzeugungskraft der Daten verloren geht und der hohe Aufwand mehr oder minder umsonst aufgebracht wird.

Damit zeigt Python einige Nachteile für Unternehmer auf, die lediglich Daten präsentieren wollen, aber nicht das nötige Hintergrundwissen für die Benutzung des Tools mitbringen.

Strengths <ul style="list-style-type: none"> • Umfangreiche Bibliotheken und Dokumentationen • Individualisierbarkeit von Dashboards 	Opportunities <ul style="list-style-type: none"> • Integration in andere Software möglich
Weaknesses <ul style="list-style-type: none"> • Keine Standardisierung von Dashboards 	Threats <ul style="list-style-type: none"> • Verlust der Überzeugungskraft der Daten durch fehlende Ästhetik

3.2. Qlik

3.2.1.Data Scientist

Um einem potenziellen Data Scientist verschiedene Optionen für den Einsatz eines Datenvisualisierungstools geben, soll nun noch eine SWOT-Analyse für das vorherige betrachtete Qlik präsentiert werden.

Ansprechende Aspekte an Qlik für einen Data Scientist dürften definitiv darin beruhen, dass es wirklich benutzerfreundlich und übersichtlich ist. Selbst wenn zu spezifischen Aspekten Unklarheiten aufkommen sollten, können diese in den meisten Fällen mit Hilfe von Online-Tutorials und Dokumentationen beseitigt werden. Zu beinahe jedem Problem können Entscheidungsfindungen durch das Internet herangezogen werden.

Was während der Test-Nutzung für die Tool-Analyse auch als Vorteil aufgetreten ist, ist die Leichtigkeit, mit welcher bei bereits gecleanteten Datensätzen verschiedene Modellierungen erstellt und auch die einzelnen Dimensionen und Ebenen verändert werden können.

Dadurch entsteht die Möglichkeit ohne viel Aufwand schnell und direkt Visualisierungen zu erstellen und diese auch individuell anzupassen. Dies ist durchaus leichter vorzunehmen als in Python.

Allerdings ist ein Data Scientist aber auch an einer möglichst großen Auswahl an Visualisierungselementen interessiert, da er möglichst breitgefächert auftreten will. Zwar konnten bezüglich dem Thema Human Development alle Fragestellungen mit den vorhandenen Visualisierungsarten bearbeitet werden, dennoch können einige Modelle innerhalb des Tools vermisst werden. So sind beispielhaft der Violin-Plot, der Pair-Plot und auch die Korrelationsmatrix nicht nutzbar.

Im Vergleich bietet Python durch die Erweiterbarkeit der vielen Bibliotheken eine deutlich größere Palette an Möglichkeit, sodass beim Einsatz von Qlik auf einige Grafiktypen verzichtet werden muss.

Weiter ist für den Data Scientist aber auch die fehlende Möglichkeit für das Data Cleaning und die Data Preparation ein Nachteil, da Datenmengen, welche Inkonsistenzen beinhalten, somit weniger gut bearbeitet werden können. Dies führt folgendermaßen zu schlechteren Visualisierungen und dementsprechend auch einem geringeren Nutzwert für den Endanwender.

Außerdem ist die kostenlose Test-Version von Qlik nur für 30 Tage gültig, wodurch die Gefahr besteht, dass längerfristige Projekte nur mit dem Einsatz dieses Tools und ohne den Einsatz von zusätzlichen Kosten nicht möglich sind. Für einen Data Scientist, der womöglich noch privat mit dem Tool arbeiten will, sind 30 US-Dollar pro Monat nämlich schon eine beträchtliche Summe.

Zudem macht man sich durch den Kauf des Tools auch abhängig vom Verkäufer, indem man auf die neuesten Updates und Features warten muss, während diese womöglich in anderen Tools schon vorhanden sind.

Strengths <ul style="list-style-type: none"> • Benutzerfreundlich • Übersichtlich • Online-Hilfen und Dokumentationen vorhanden • Leichte Modellierung von einfachen Grafiken 	Opportunities <ul style="list-style-type: none"> • Ohne viel Aufwand Visualisierungen erstellen • Individualisierbarkeit im Nachhinein
Weaknesses <ul style="list-style-type: none"> • Weniger Auswahl an Grafiken als in Python • Fehlende Möglichkeiten zum Data Cleaning und zur Data Preparation 	Threats <ul style="list-style-type: none"> • Längerfristige Projekte durch 30-tägige Testversion nicht möglich • Abhängigkeiten zum Verkäufer entstehen

3.2.2. Betrieblicher Entscheider

Um für den betrieblichen Entscheider auch eine Auswahl zwischen den Tools zu ermöglichen, wird für diesen nun auch noch eine SWOT-Analyse bezüglich Qlik erstellt.

Für diesen ist es vermutlich eher weniger wichtig umfangreiche Datenvisualisierungen zu entwerfen, wie es der Data Scientist für gewöhnlich bewerkstelligt. Der betriebliche Entscheider verfolgt primär das Ziel, so effizient wie möglich Dashboards für die Entscheidungsfindung in seinem Unternehmen zu entwerfen.

Hierfür benötigt er ein übersichtliches und benutzerfreundliches Tool, mit welchem er schnell und einfach passende Visualisierungen erzeugen kann. Dies erhält er mit dem Verwenden von Qlik, indem er auf der Startseite einen Überblick über grundlegende Diagrammartentypen erhält und diese per einfachem Maus-Click auswählen kann.

Somit kann er die Stärken des Tools sehr gut ausnutzen.

Des Weiteren dürfte ihm zugutekommen, dass keinerlei Programmiererfahrungen notwendig sind, um adäquate Ergebnisse liefern zu können, wenn der importierte Datensatz bereits vollständig bereinigt ist. Wie dem Data Scientist, können hierbei auch die zahlreichen Dokumentationen im Internet dem betrieblichen Entscheider dabei helfen, bei Unklarheiten voranzukommen.

Nachteile bieten ihm sich nur dann, wenn er versucht mit einem Datensatz zu arbeiten, der Inkonsistenzen enthält. Dadurch könnten seine Visualisierungen dann unbrauchbar werden, da zu viele Fehlstellen in die erstellte Grafik eingebaut werden.

Ebenfalls könnte auch für ihn die teure Vollversion des Tools zum Problem werden, wenn das Geld für dieses nicht vorhanden ist. Wobei die finanzielle Sachlage im betrieblichen

Kontext weniger ein Problem sein dürfte als beim Data Scientist, der das Tool auch für private Zwecke benutzt.

Strengths <ul style="list-style-type: none">• Übersichtlich• Benutzerfreundlich• Online-Hilfen und Dokumentationen vorhanden• Leichte Modellierung von einfachen Grafiken	Opportunities <ul style="list-style-type: none">• bei Vollversion Möglichkeit für langfristige Projekte
Weaknesses <ul style="list-style-type: none">• Handling von mehreren Datensätzen von unerfahrenen Personen schwer	Threats <ul style="list-style-type: none">• bei Datensätzen mit Inkonsistenzen, könnte Visualisierungen unbrauchbar werden

4. Fazit

Da in der heutigen Gesellschaft, sowohl im privaten als auch betrieblichen Umfeld, Daten eine immer größere Rolle spielen, ist es wichtig auch ein geeignetes Tool für den jeweiligen Anwendungszweck zu ermitteln.

Deshalb wurde die Analyse von Python und Qlik und deren Vergleich in der Ausarbeitung zum zentralen Bestandteil.

Als abschließendes Fazit zu jener Toolanalyse, kann festgehalten werden, dass beide Tools für jeweilige spezifische Anwendungsfälle ihre Vor- und Nachteile besitzen.

Python hat sich im Laufe der Analyse als Tool herausgestellt, in welchem die Möglichkeiten zur Datenvisualisierung durch den Einsatz verschiedener Bibliotheken wie „plotly“ oder „pandas“ nahezu unbegrenzt sind.

Mannigfaltige Varianten an Modellierungsarten können hierfür ausgewählt werden.

Sind die nötigen Programmiererfahrungen beim jeweiligen Endanwender vorhanden, so kann mit umfangreichen Datenmengen gearbeitet und durch die Möglichkeit von Datenbereinigung und Datenvorbereitung, für die letztendliche Verwendung, auch ein präzises Visualisierungsergebnis erzielt werden.

Mögliche Anwendungsszenarien für Python könnten, aufgrund der freien Verfügbarkeit ohne zusätzliche Kosten, die Privatanwendung bei Personen, die sich schlichtweg für den Umgang mit Daten interessieren, aber auch die Nutzung in Unternehmen sein. Gerade im unternehmerischen Kontext, in welchem tagtäglich mit großen Datenmengen gearbeitet wird, benötigt es ein flexibel einsetzbares Tool, das weitreichende Ergebnisse in Bezug auf Daten liefern kann. Dafür ist Python auf jeden Fall sehr gut geeignet, da Unternehmen auch meistens Mitarbeiter beschäftigen, die genau auf solche Anwendungsfälle spezialisiert sind und das nötige Hintergrundwissen besitzen, um die verschiedenen Befehle und Bibliotheken einzusetzen.

Als Vergleichsobjekt zu Python wurde auch Qlik analysiert. Das cloudbasierte Tool zeichnet sich vor allem durch seine übersichtliche Oberfläche, in welcher leichte Visualisierungen ohne geringen Aufwand erstellt und auch individualisiert werden können, aus. Im Gegensatz zu Python bedarf es hierbei keinerlei Erfahrung im Umgang mit Programmiersprachen, sodass praktisch jeder es benutzen kann.

Allerdings stellen die hohen monatlichen Kosten eine Barriere für die Nutzung von Privatpersonen dar. In dieser Hinsicht kann Python mit umfassenderen Möglichkeiten im Umgang mit Daten bevorzugt werden. Vor allem als Data Scientist außerhalb des geschäftsbetrieblichen Prozesses, eröffnen sich außer dem leichten Erstellen für einfache Visualisierungen kaum Vorteile, welche die monatliche Bezahlung von 30 US-Dollar rechtfertigen.

Nichtsdestotrotz ist Qlik für den betrieblichen Kontext eine Option, in welchem es ein Anwendungsszenario finden kann.

Ein Beispiel hierfür könnte ein Angestellter sein, welcher keine Programmiererfahrung besitzt, aber dennoch einfache Visualisierungen für eine Präsentation konzipieren möchte. Ohne größeren Aufwand kann ein anschauliches Gesamtbild erzeugt werden, was seinen Zweck erfüllt.

Da auch die Lizenzkosten für ein wirtschaftlich gut aufgestelltes Unternehmen kein Problem darstellen sollten, stellt dies ein mögliches Anwendungsszenario dar.

Somit ist die Gesamtanalyse von Python und Qlik mit Blick auf verschiedenste Kriterien und dem Aufzeigen von verschiedenen Anwendungsszenarien abgeschlossen.