R-Integration in Qlik® Produkten (QlikView®/ Qlik Sense®)

Whitepaper, Juli 2018

Autoren:

Jonas Stopschinski, Data Scientist
David Stopschinski, Data Scientist
Lukas Seppelfricke, Consultant

www.evaco.de

Wofür braucht man R?

Was ist R und welche Vorteile ergeben sich daraus in der Verwendung von QlikView®/Qlik Sense®?

R ist eine Skriptsprache, die zum Programmmieren von statistischen Berechnungen entwickelt wurde. R ist auf Basis der General Public License (GPL) entwickelt, sodass es kostenlos von der Webseite (https://www.r-project.org/) geladen werden kann. Es steht eine ganze Reihe an statistischen Techniken wie unter Anderem lineare und nichtlineare Modellierung, klassische statistische Tests sowie Zeitreihenanalyse und Clustering zur Verfügung. Durch die R-Community kann jeder aus dem großen Repertoire zu fast jedem Thema ein passendes Paket finden und sogar eigene Pakete erstellen. Diese Möglichkeiten machen R sehr flexibel.

Wie wird R in QlikView®/Qlik Sense® integriert?

In diesem Fall sprechen wir von einer Server-Side-Extension (SSE). Damit kann die Programmbibliothek der Qlik® Produkte um die Bibliothek der R-Befehle erweitert werden. Das kann sowohl im Ladeskript als auch in Diagrammen auf der graphischen Oberfläche geschehen.

Ob man die Funktionalität von R im Skript oder an der graphischen Oberfläche verwendet, hängt vom jeweiligen Anwendungsfall ab. Möchte man einen statischen Datensatz mit R erzeugen (der Datensatz soll einmal erstellt und nicht mehr verändert werden), den man dann an der Oberfläche verwendet, bietet sich die Integration der R-Skripte im Ladeskript von QlikView® bzw. Qlik Sense® an. Möchte man aber das Modell nach jeder erneuten Selektion dynamisch anpassen, kann man die R-Funktionen in den Diagrammen an der graphischen Oberfläche nutzen. Das Prinzip funktioniert folgendermaßen: Die zu verarbeitenden Datensätze werden an R geschickt (R kann auch auf einem physisch unabhängigen Server installiert sein), dort werden die Daten verarbeitet und Berechnungen durchgeführt und schließlich werden die Ergebnisse an QlikView® bzw. Qlik Sense® zurückschickt, wo die Daten dann wie gewohnt als Diagramm dargestellt werden.

Die Engine funktioniert dynamisch, so wie man es von Qlik® Produkten gewohnt ist. Das bedeutet, dass der Nutzer z.B. eine neue Auswahl in einem Feld vornehmen kann und das Ergebnis adhoc mit R berechnet und angezeigt wird. Somit ändert sich nichts für den Dashboardnutzer: die gewohnte schnelle Anpassung an eine neue Auswahl bleibt bestehen. Und trotzdem: es gibt eine Fülle neuer Möglichkeiten, Ihre Daten zu analysieren und zu modellieren.

Für wen wurden diese Funktionen entwickelt?

Um komplexe Dashboards mit dieser neuen Technologie erstellen zu können, sollte bereits Wissen sowohl in R als auch in QlikView® bzw. Qlik Sense® vorhanden sein. Für einige „Standardszenarien“ gibt es bereits Templates, mit denen man ohne großes Vorwissen Analysen durchführen kann.

R kann in Qlik® Produkten benutzt werden, um verschiedene statistische Analysen durchzuführen. Die Möglichkeiten sind nur beschränkt durch die Funktionalitäten, die es in R gibt. Es besteht uneingeschränkter Zugang zu allen R-Paketen, die die R-Community bieten kann. Ebenso besteht uneingeschränkter Zugang zu allen Funktionen und Möglichkeiten, die Qlik® zu bieten hat. Statistische Berechnungen, die damit durchgeführt werden können, sind (unter Anderem):

- **Clusteranalysen:** z. B. k-means – eine statistische Methode, um Muster in einem Datensatz zu finden.
- **Simulationen:** Methode, um Konsequenzen verschiedener Szenarien durchschauen zu können. Z.B. kann eine Monte Carlo Simulation erstellt werden, um verschiedene Szenarien für den Preis von Öl auf die Entwicklung der eigenen Firma anwenden zu können
- **Zeitreihenanalyse:** Um ein aussagekräftiges Ergebnis mit einer Zeitreihe zu erhalten, müssen Daten mit dem richtigen Modell dargestellt werden. Für R gibt es Pakete, um eine automatische Auswahl für ein ARIMA-Modell zu finden.

Diese Liste ist natürlich nicht vollständig. Es gibt Pakete in R für fast alle statistischen Berechnungen, sodass es selten nötig ist, komplizierte Algorithmen selber zu erstellen. Sollten die Pakete nicht ausreichen, kann man natürlich beliebig komplexe Algorithmen selbst entwickeln und in QlikView® bzw. Qlik Sense® benutzen.

Beispiel eines Forecasts mit Hilfe des ARIMA-Modells

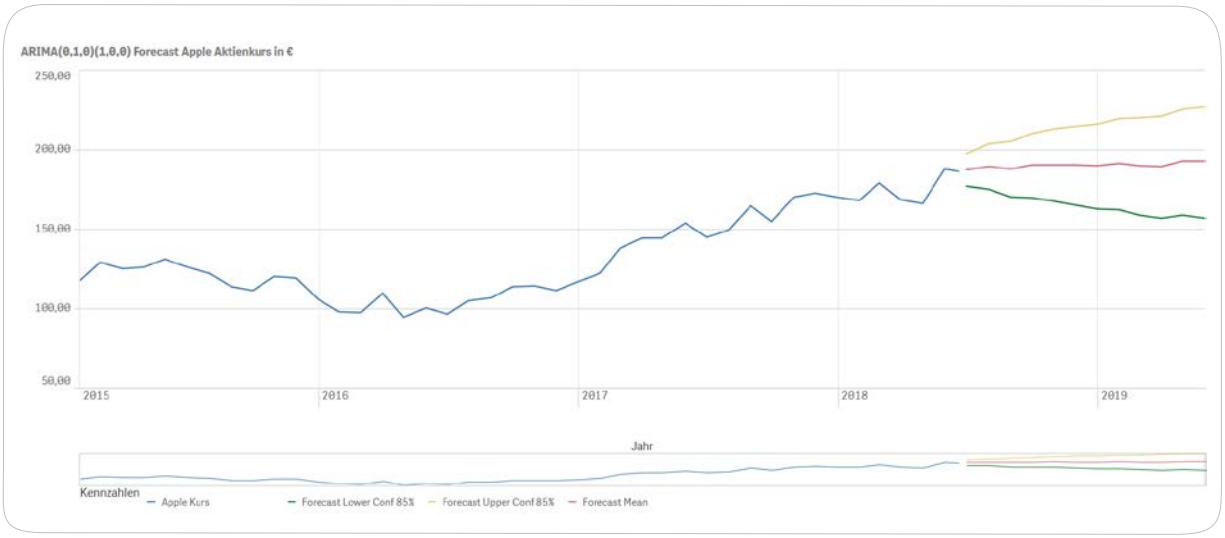

Abbildung 1: ARIMA(0,1,0)(1,0,0)

In diesem Beispiel wird ein Forecast des Apple Aktienkurses mit Hilfe eines ARIMA(0,1,0)(1,0,0)-Modells erstellt und erläutert.

ARIMA steht für **A**uto**R**egressive **I**ntegrated **M**oving **A**verage und hat in diesem Beispiel die folgende Form:

$$X_t = \epsilon_t + X_{t-1} + \phi_{12} * X_{t-12} - \phi_{12} * X_{t-13} .$$

Allgemein hat das Modell die Definition ARIMA(p,d,q)(P,D,Q) mit den 6 Parametern p,d,q,P,D,Q. Die kleinen Buchstaben stehen dabei für „nicht-saisonale“ und die großen Buchstaben für „saisonale“ Komponenten im Modell.

Das „p“ ist der Grad der **A**uto**R**egressiven Terme (AR-Terme); das „d“ entsprechend für Grad der Integration und „q“ für den Grad der **M**oving **A**verage Terme (MA-Terme). Die Buchstaben „P“,“D“ und „Q“ stehen entsprechend für die saisonalen AR-Terme (SAR-Terme), die saisonale Integration und die saisonalen MA-Terme (SMA-Terme). In diesem Modell haben wir die Saisonalität auf ein Jahr (also 12 Monate) gesetzt. Entsprechend bedeutet ein ARIMA(0,1,0)(1,0,0)-Modell , dass sich die Entwicklung des Aktienkurses von Apple aus einem Random Walk und einem saisonalen Autoregressiven Term zusammensetzt.

Mathematisch bedeutet das:

$$(1 - \phi_{12}B^{12})(1 - B)X_t = \epsilon_t .$$

Multipliziert man den Term aus, erhält man:

$$(1 - \mathrm{B} - \phi_{12}B^{12} + \phi_{12}B^{13})X_t = \epsilon_t.$$

Auflösen nach $\mathrm{X_t}$ ergibt schließlich:

$$\mathrm{X_t} = \epsilon_t + X_{t-1} + \phi_{12} * X_{t-12} - \phi_{12} * X_{t-13}$$

Diese Gleichung ist das Modell für Forecasts mit dem kleinsten AIC (=**A**kaike **I**nformation **C**riterion; ein Kriterium, um die Qualität eines Modells zu schätzen). Ausgewählt wurde eine Kombination aus allen möglichen Kombinationen der Werte p,d,q,P,D,Q.

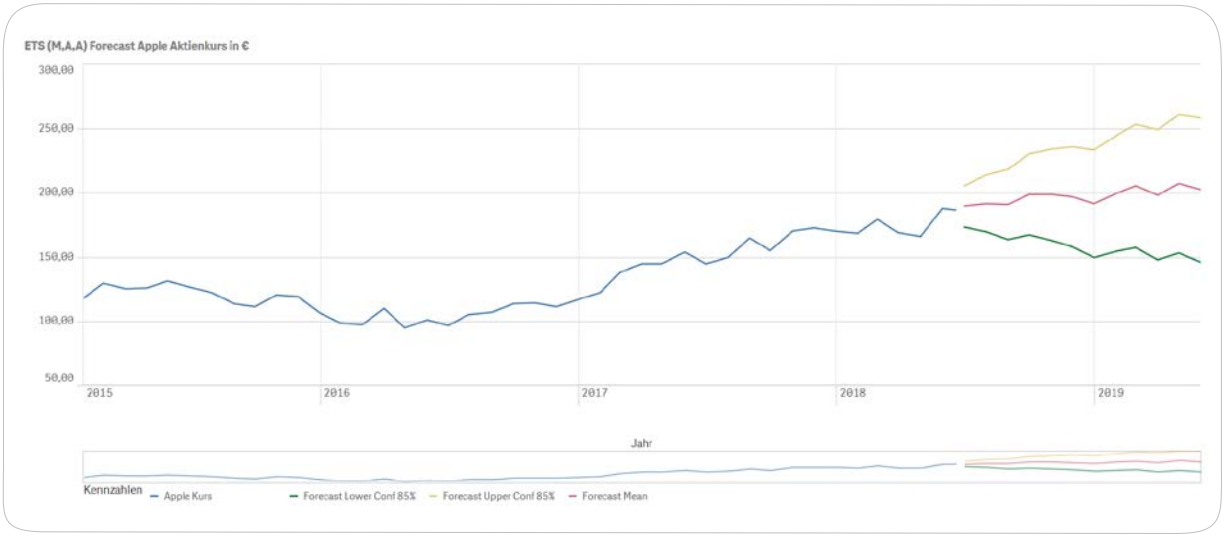

Abbildung 2: ETS(M,A,A)

ARIMA ist nicht die einzige Klasse von Forecast-Modellen. Eine weitere, sehr wichtige Forecast-Klasse, die für Zeitreihen geeignet sind, sind die ETS-Modelle. Diese Modelle teilen den Prozess in drei Komponenten auf: 1) Error, 2) Trend und 3) Saisonalität.

In allen drei Fällen kann zwischen „N"=keine, „A"=additiv, „M"=multiplikativ und „Z"=automatisch ausgewählt werden.

In diesem Beispiel wurden folgende Kriterien genommen:

Error: multiplikativ,

Trend: additiv,

Saisonalität: additiv.

Daraus ergibt sich folgendes Modell in Komponentenschreibweise:

Prognosegleichung:

$$X_t = l_{t-1} + b_{t-1} + s_{t-m} + \epsilon_t$$

Die einzelnen Komponenten in den Summanden lassen sich rekursiv definieren:

1.) l_t: Levelkomponente der Zeitreihe:

$$l_t = l_{t-1} + b_{t-1} + \alpha\epsilon_t$$

2.) b_t: Trendkomponente der Zeitreihe:

$$b_t = b_{t-1} + \beta\epsilon_t$$

3.) s_t: Saisonkomponente der Zeitreihe:

$$s_t = s_{t-m} + \beta\epsilon_t$$

4.) ϵ_t: Multiplikativer Fehler:

$$\epsilon_t = (X_t - \hat{X}_t)/\hat{X}_t$$

In den Abbildungen 1 und 2 ist der Forecast als rote Linie gekennzeichnet. Die 85%-Vertrauensintervalle sind als gelbe und grüne Linie zu sehen.