Datenbanksysteme - Abgabe DWH

Lukas Forsthuber, Peter Jernej, Christian Brandstätter

Inhalt

[Kurzübersicht Workflow 1](#_Toc40129822)

[Installation Pentaho 1](#_Toc40129823)

[Auswahl und Kurzbeschreibung Datenset 2](#_Toc40129824)

[Transformation Datenset in Sternschema (Python) 4](#_Toc40129825)

[Import in Pentaho und Schreiben der Tabellen in postgres Datenbank 4](#_Toc40129826)

[Mehrdimensionale Analyse Datenset 8](#_Toc40129827)

# Kurzübersicht Workflow

1. Installation JDK, Pentaho, postgres
2. Herunterladen Sales-Datensatz
3. Vorbearbeitung Datensatz mit Python -> Sternschema
4. Import in Pentaho
5. Übertragen in postgres-db
6. Re-Import von postgres-db
7. Mehrdimensionale Analyse pentaho
8. Datenexport -> csv
9. Darstellung der Ergebnisse in R

# Installation Pentaho

Für den Betrieb von Pentaho wurde das Betriebssystem Windows 10 verwendet.

Die Pentaho–Suite wurde heruntergeladen von Sourceforge[[1]](#footnote-1). Für den Betrieb wurden auch Java 8 inkl. JRE und JDK[[2]](#footnote-2) installiert. Danach wurde das System neu gestartet, um die Systempfade zu aktualisieren.

Vor der ersten Verwendung von Pentaho wurde die Datei set-pentaho-env.bat (Pentaho-Stammverzeichnis) ausgeführt. Die Pentaho-Suite lässt sich mit der Datei spool.bat im Pentaho-Stammverzeichnis öffnen. Zusätzlich wurden Postgres 12.2 und pgAdmin 4.20 installiert[[3]](#footnote-3).

Zum Importieren der CSV-Dateien waren noch weitere Schritte erforderlich:

Installieren eines Plugins im Markplatz: Import from txt. Danach musste ein Pentaho-Neustart durchgeführt werden.

# Auswahl und Kurzbeschreibung Datenset

Auswahl:

Link: <https://www.kaggle.com/kyanyoga/sample-sales-data#sales_data_sample.csv>

Es handelt sich beim Datenset um ein künstliches Verkaufsdatenset. Im Folgenden ein Zitat von Kaggle zum Datenset:

Sample Sales Data, Order Info, Sales, Customer, Shipping, etc., Used for Segmentation, Customer Analytics, Clustering and More. Inspired for retail analytics. This was originally used for Pentaho DI Kettle, But I found the set could be useful for Sales Simulation training.

Originally Written by María Carina Roldán, Pentaho Community Member, BI consultant (Assert Solutions), Argentina. This work is licensed under the Creative Commons Attribution-Noncommercial-Share Alike 3.0 Unported License. Modified by Gus Segura June 2014.

Das Datenset liegt als csv vor. Die Charakter-Enkodierung ist latin-1. Das Datenset enthält 25 Spalten und 2823 Zeilen. Details zu den Variablen werden in der nächsten Tabelle dargestellt (nächste Seite). Offenbar gab es bei der Daten-Erstellung einen Fehler, die Variable Priceeach hatte bei 100 eine Obergrenze. Das wurde über die offenkundige Beziehung Sales = PRICEEACH \* QUANTITYORDERED korrigiert.

Zusätzlich wurde ein Primary Key erzeugt, durch die Zusammenfügung von ORDERNUMBER und ORDERLINENUMBER.

Die Datei wurde in ein Stern-Schema umgewandelt. Dabei wurden die zentralen Variablen, die die Bestellungen (Orders) betreffen, im Zentrum (Dimension center) belassen (siehe nächste Tabelle). Um die Daten aufzuteilen, wurde in Python eine Funktion definiert, die für Unique-Entries aufsteigende Integersequenzen erzeugt, die sich auch mit einem Text kombinieren lassen.

Nach unzähligen Fehlversuchen des csv-Imports in Pentaho wurde der Zwischenweg über Python ausgewählt. Es gab laufend irgendwelche Daten-Typ-Probleme, zu Beginn auch wegen der Datei-Enkodierung. Nachdem alle Datentypen auf String gesetzt wurden, verschwanden diese zwar, das hätte aber andere negative Folgen nach sich gezogen.

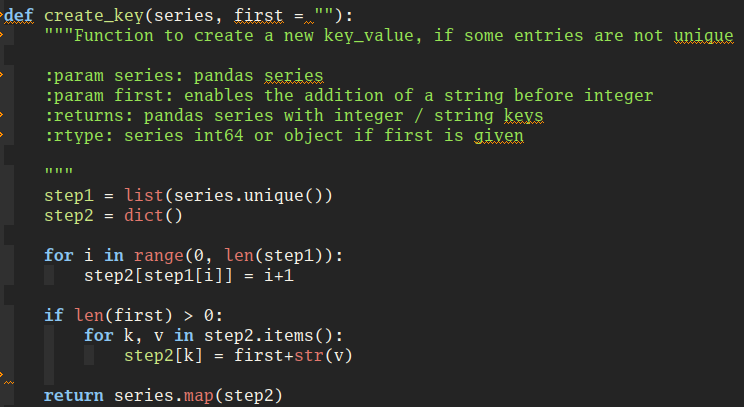


Abbildung: Python-Funktion zum Definieren von Keys

|  |  |  |  |
| --- | --- | --- | --- |
| **Variable** | **Anzahl befüllter Zellen** | **Datentyp** | **Dimension** |
| ORDERNUMBER | 2823 | int64 | **Center** |
| QUANTITYORDERED | 2823 | int64 | **Center** |
| PRICEEACH | 2823 | float64 | **Center** |
| ORDERLINENUMBER | 2823 | int64 | **Center** |
| SALES | 2823 | float64 | **Center** |
| ORDERDATE | 2823 | object | **Ordertimes** |
| STATUS | 2823 | object | **Center** |
| QTR\_ID | 2823 | int64 | **Ordertimes** |
| MONTH\_ID | 2823 | int64 | **Ordertimes** |
| YEAR\_ID | 2823 | int64 | **Ordertimes** |
| PRODUCTLINE | 2823 | object | **Products** |
| MSRP | 2823 | int64 | **Products** |
| PRODUCTCODE | 2823 | object | **Products** |
| CUSTOMERNAME | 2823 | object | **Customer** |
| PHONE | 2823 | object | **Customer** |
| ADDRESSLINE1 | 2823 | object | **Customer** |
| ADDRESSLINE2 | 302 | object | **Customer** |
| CITY | 2823 | object | **Customer** |
| STATE | 1337 | object | **Customer** |
| POSTALCODE | 2747 | object | **Customer** |
| COUNTRY | 2823 | object | **Customer** |
| TERRITORY | 1749 | object | **Customer** |
| CONTACTLASTNAME | 2823 | object | **Customer** |
| CONTACTFIRSTNAME | 2823 | Object | **Customer** |
| DEALSIZE | 2823 | object | **Center** |

# Sternschema (Python)

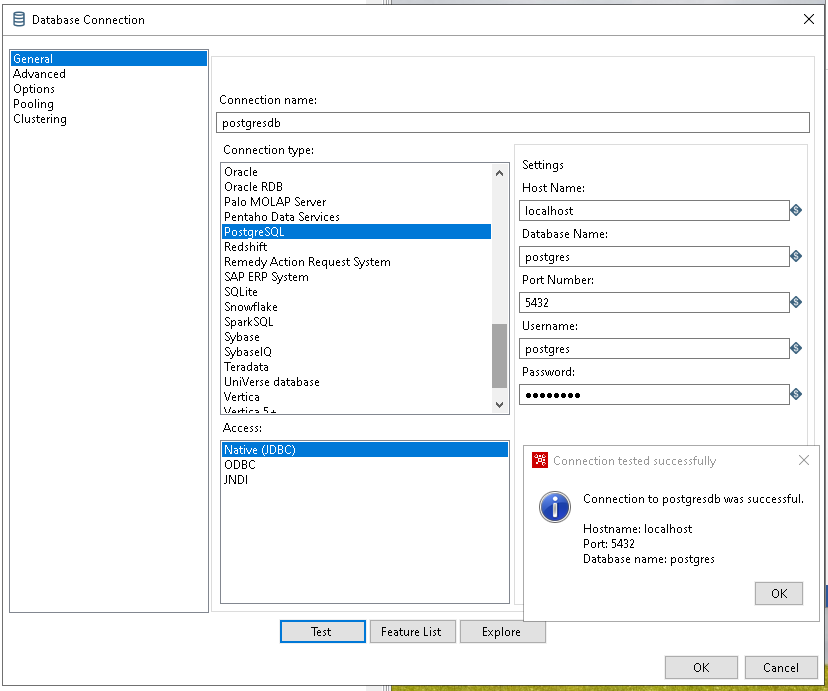
Für die Aufteilung in das Sternschema in die Dimensionen wurden verschiedene Schlüssel erzeugt. In der Zentraltabelle wurden die Schlüssel hinterlegt. PK (Primary Key) enthält die Ordernumber, einen Unterstrich und die Orderlinenumber, ON\_ID basiert auf der Ordernumber (notwendig für die Ordertimes, da diese ja mit den Orders zusammenhängen), PR\_ID basiert auf dem Productcode und CU\_ID auf dem Customername.

Die Dimension Customer enthält auch die räumliche Dimension, da diese in dem verwendeten Datensatz über die Kunden einbezogen wird. In dem gegenständlichen einfachen Datenschema ergibt sich die Verknüpfung mit den Dimensionen über die einzelnen Keys.

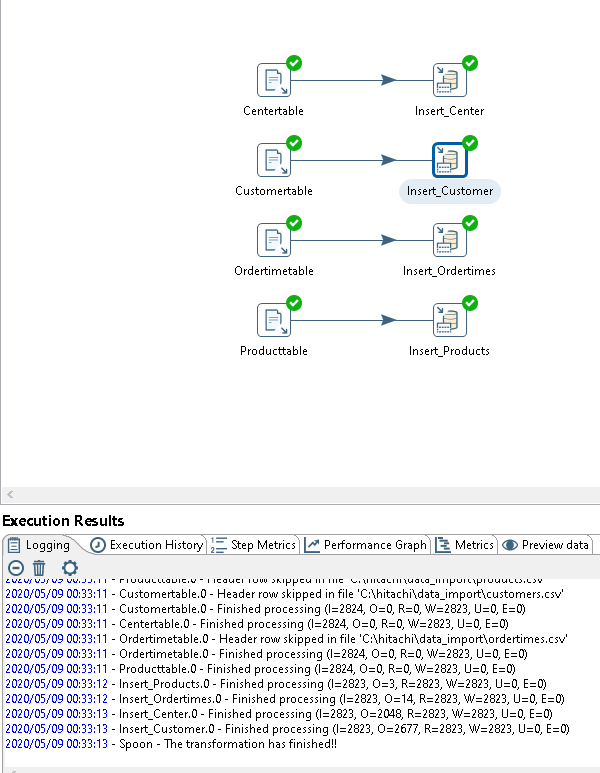
|  |  |
| --- | --- |
| **Schlüssel** | **Dimension** |
| ON\_ID | Ordertimes |
| CU\_ID | Customer (auch räumlich) |
| PR\_ID | Product |

# Import in Pentaho und Schreiben der Tabellen in postgres Datenbank

Durch die Vorbereitung in Python, war der Datenimport und die Übertragung in die Datenbank dann relativ einfach möglich.



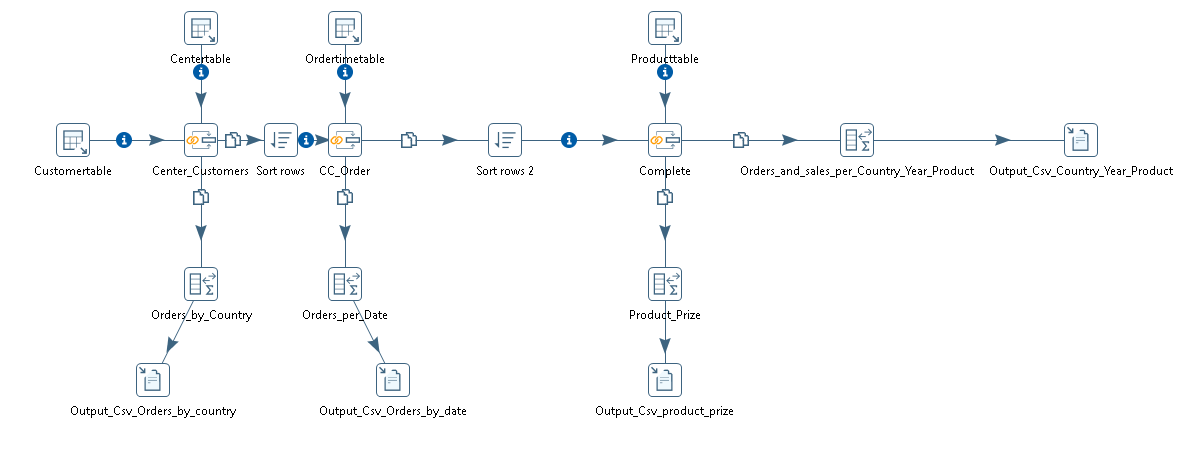
Der Übertragungsprozess hat nach der Anpassung der Spaltengrößen funktioniert.



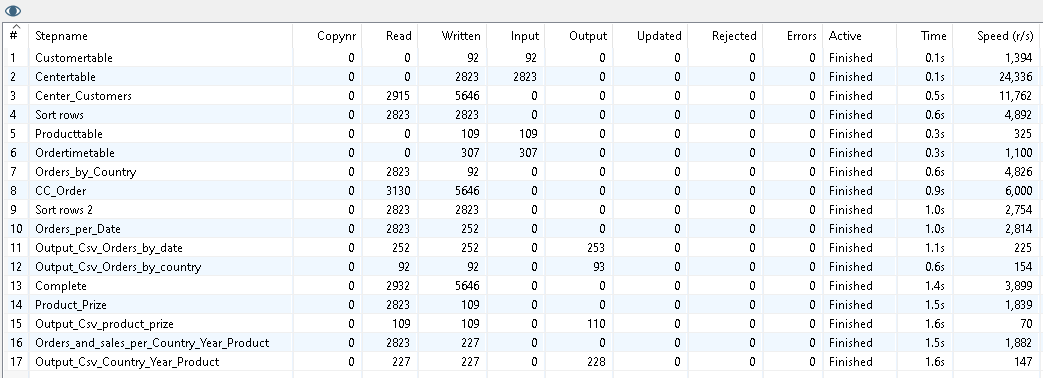
Sternschema

# Mehrdimensionale Analyse Datenset

Der Olap-Prozess wurde mehrstufig abgebildet. Zuerst wurden die Kundendaten zur Zentraltabelle gejoined, dann die zeitliche Dimension und abschließend die Produktdetails. Das ermögliche die mehrdimensionale Betrachtung im Prozessverlauf.



Um die Daten fehlerfrei verknüpfen zu können (inner joins), mussten diese vorher aufsteigend sortiert werden (Sort rows und Sort rows 2). Die eigentliche Analyse passiert mit den Group\_by-Memory-Operation, wo verschiedene Variablen unterschiedlicher Join-Stufen zusammen aggregiert werden konnten.



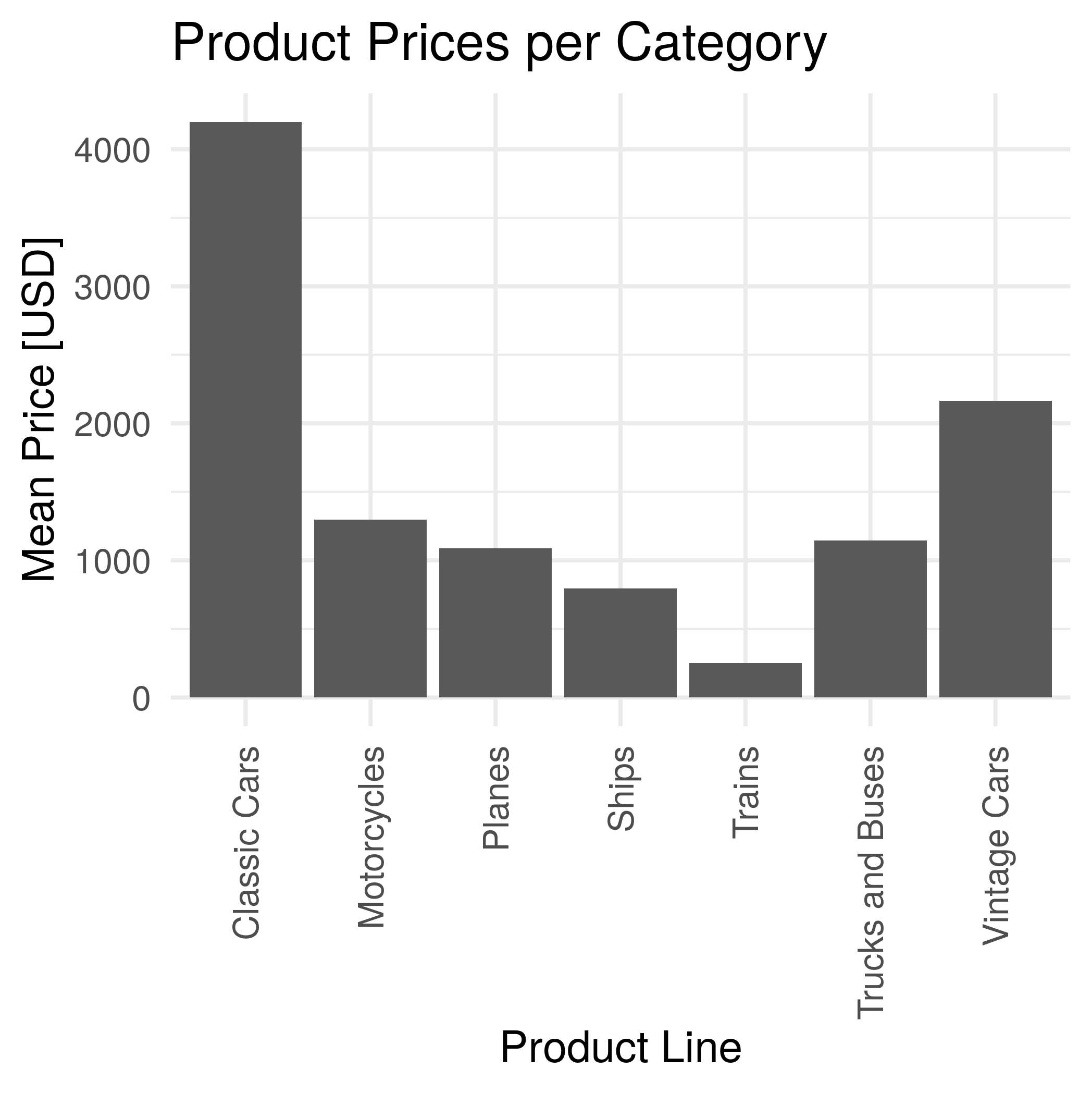
Bei der Performance (Abbildung Step Metrics) sieht man, dass vor allem die joins (Center\_Customers, CC\_Order, Complete), deren Vorsortierung (Sort rows und Sort rows 2) und die CSV-Exporte relativ viel Zeit benötigten. Es wird auch ersichtlich, dass, je weiter rechts ein Prozess in der Prozesskette steht, dieser eher mehr Zeit benötigt.

# Ergebnisse

Wir haben uns folgende Fragen zum Datenset überlegt:

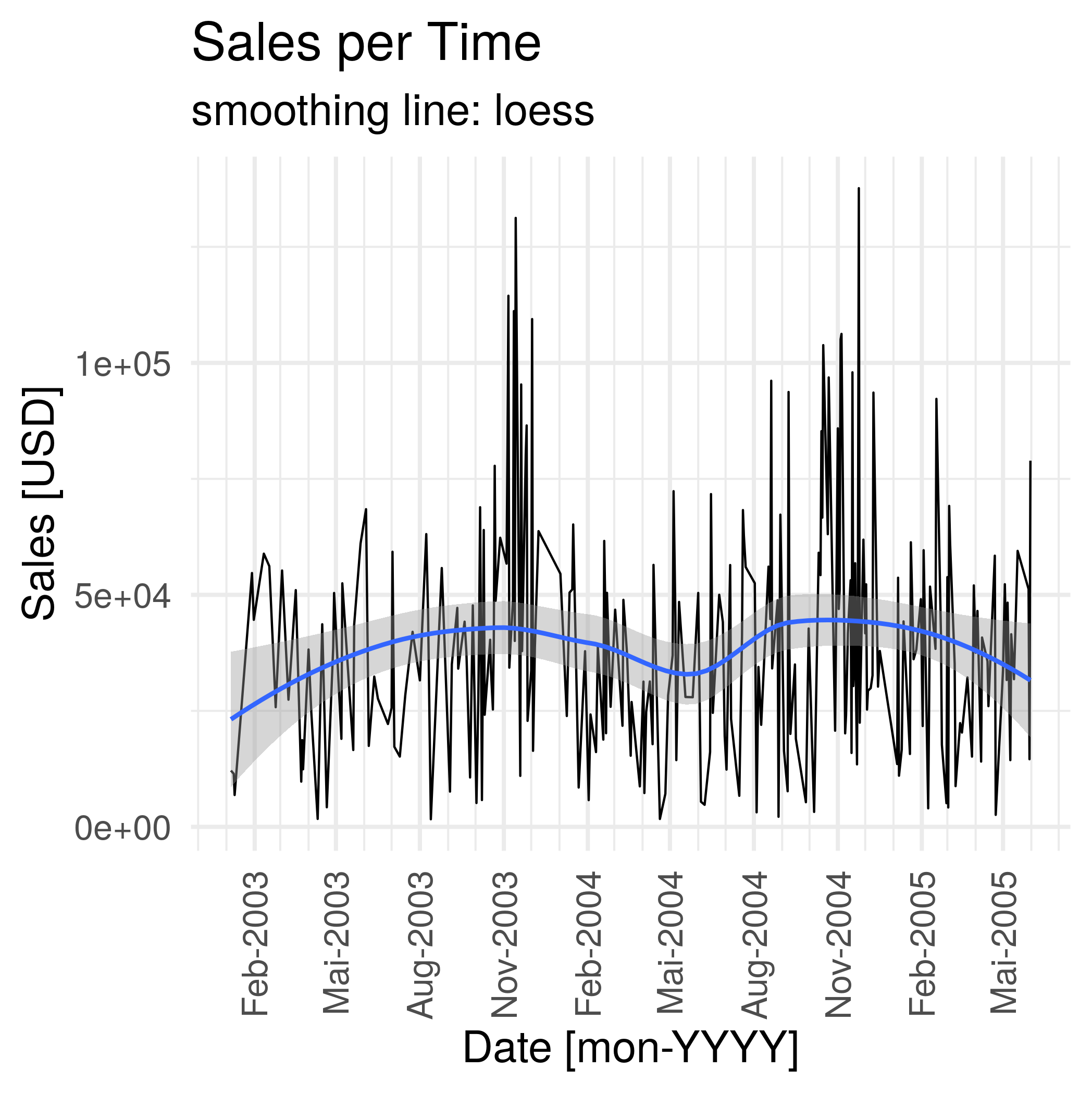
1. Welche Produkte sind am ertragreichsten?
2. Wie verlaufen die Verkäufe über die Zeit?
3. In welchen Ländern wird am meisten verkauft?
4. Wer sind die wichtigsten Kunden und wo sitzen diese?
5. Welche Produkte werden wann und wo am meisten verkauft?
6. Welche Produkte sind am ertragreichsten?

Die höchsten Preise werden für Classic Cars erzielt, gefolgt von Vintage Cars. Natürlich heißt dass nicht, dass diese Produkte auch den höchsten Gewinn erzielen…



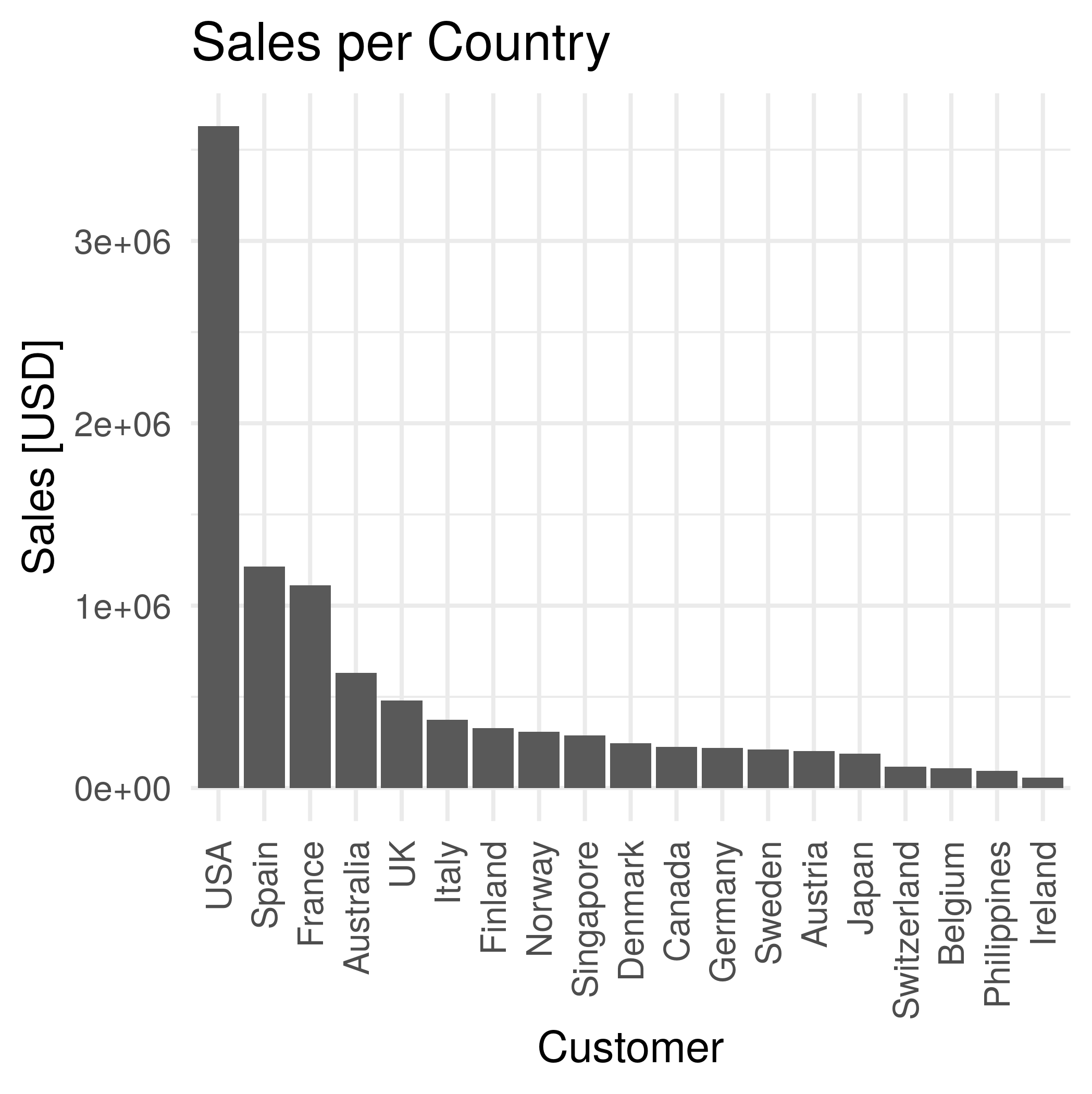
1. Wie verlaufen die Verkäufe über die Zeit?

Offensichtlich gibt es vor Weihnachten große Peaks in den Verkaufszahlen.



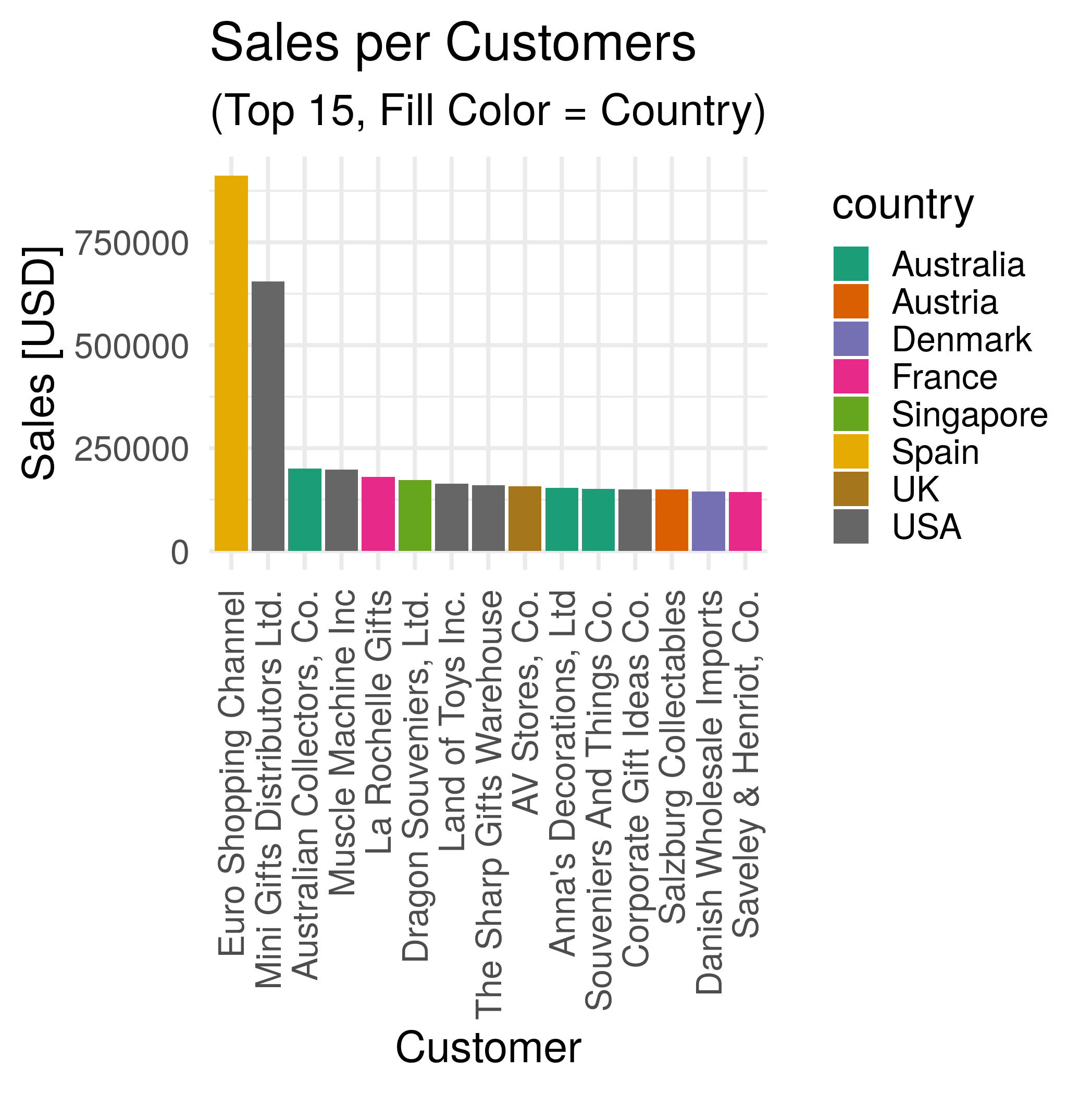
1. In welchen Ländern wird am meisten verkauft?

Offenbar wurde in den USA, gefolgt von Spain und France am meisten abgesetzt.



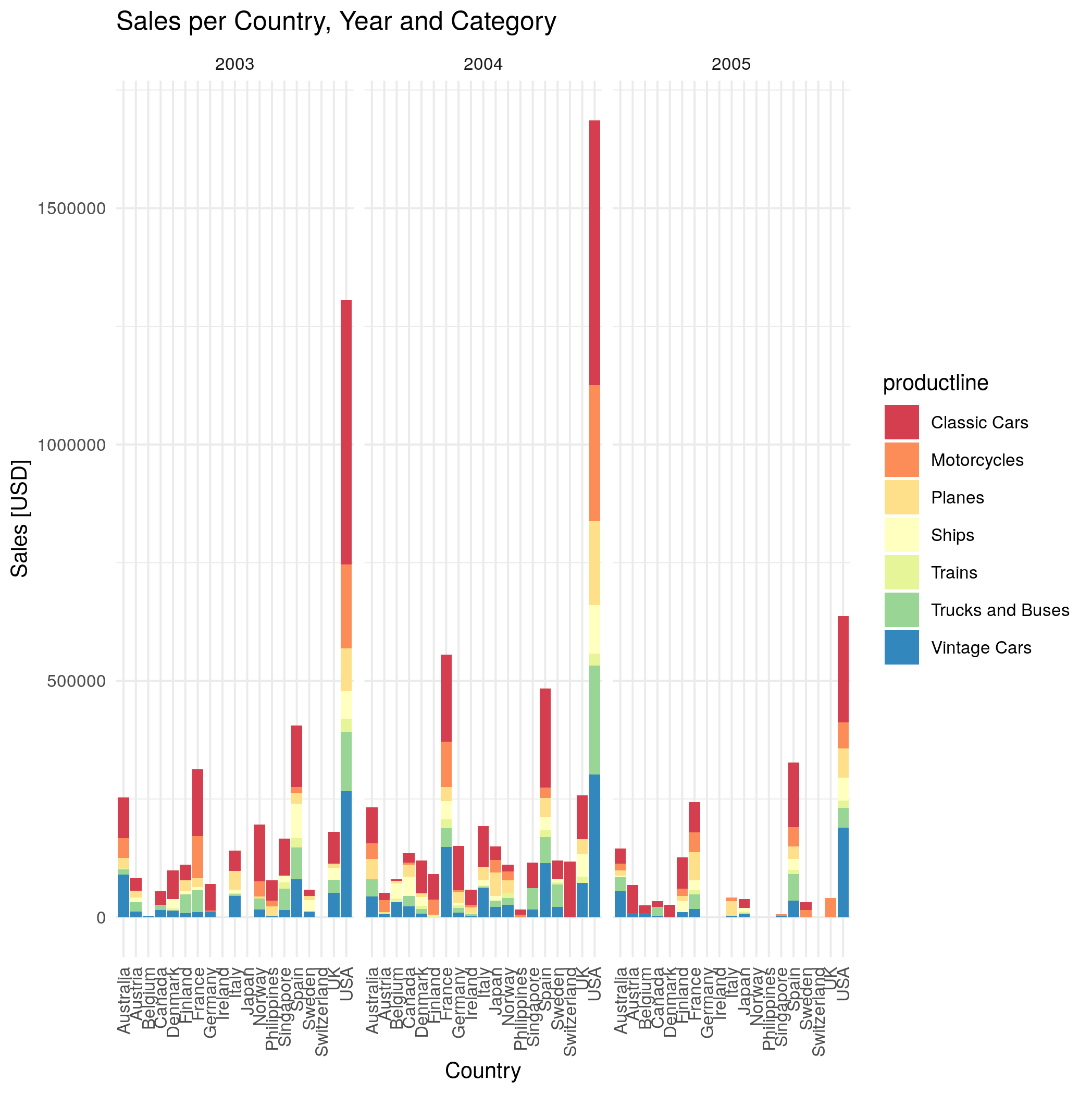
1. Wer sind die wichtigsten Kunden und wo sitzen diese?

Die Top 3 sind Euro Shopping Channel (Spain), Mini Gifts Distributors Ltd. (USA) und nach einem starken Abfall Australian Collectors, Co. (Australia).



1. Welche Produkte werden wann und wo am meisten verkauft?

Vor allem in den USA wurden viele Classic Cars verkauft, besonders im Jahr 2004. Generell dürften Classic Cars und Vintage Cars am meisten verkauft werden, was vernünftig ist, da diese die höchsten Preise erzielen. 2005 war für alle Länder ein schlechtes Jahr.



1. <https://sourceforge.net/projects/pentaho/files/Pentaho%209.0/>; April 2020 [↑](#footnote-ref-1)
2. <https://www.oracle.com/java/technologies/javase-jdk8-downloads.html> (jdk-8u251-windows-x64), April 2020 [↑](#footnote-ref-2)
3. <http://www.enterprisedb.com/thank-you-downloading-postgresql?cid=48>, April 2020 [↑](#footnote-ref-3)