Introduction to Tableau

Fernando Birra
João Moura Pires

Notice

- **Author**
  - **João Moura Pires (jmp@fct.unl.pt)**
  - **Fernando Birra (fpb@fct.unl.pt)**

-

-

Table of Contents

- **What is Tableau**
- **Tableau Software Overview**
- **Main Concepts**

This presentation includes video fragments from this video

Getting Started - Tableau

What is Tableau

What is Tableau

- **Tableau is the name of a company specialized in data products.**

What is Tableau

- **Tableau is the name of a company specialized in data products.**
- **Mission statement taken from tableau.com:**

In 2020 the world will generate 50 times the amount of data as in 2011 and 75 times the number of information sources (IDC, 2011).

Tableau helps people see and understand data

What is Tableau

- **Tableau is the name of a company specialized in data products.**
- **Mission statement taken from tableau.com:**

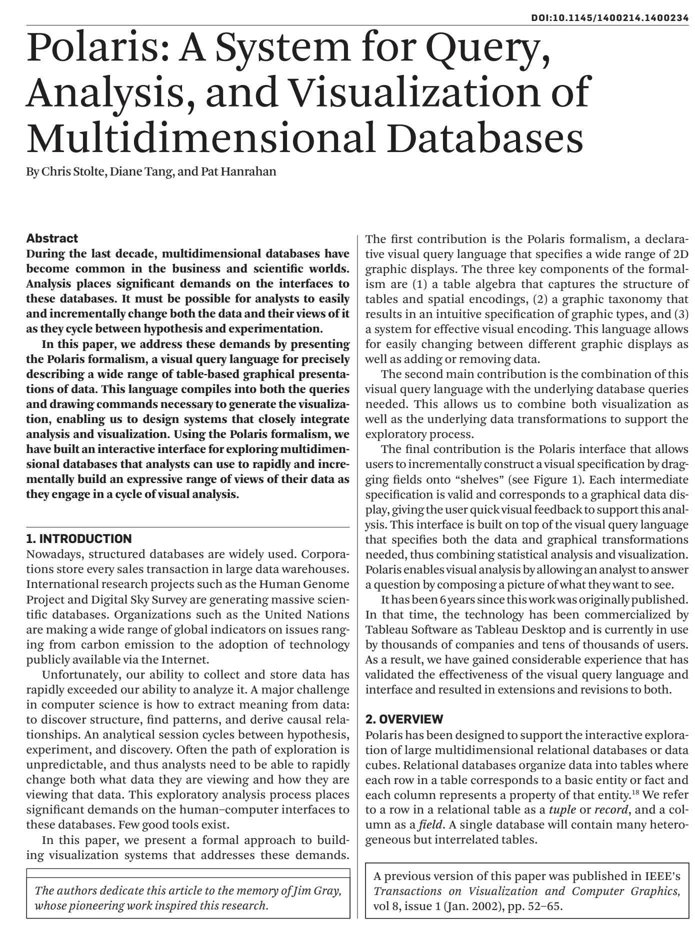
> *In 2020 the world will generate 50 times the amount of data as in 2011 and 75 times the number of information sources (IDC, 2011).*
>
> ***Tableau helps people see and understand data***

- **"It started with a brilliant computer scientist (Chris Stolte), an Academy-Award winning professor (Pat Hanrahan) and a savvy business leader (Christian Chabot)".**
- **Polaris - Interactive database visualization was a project developed at Stanford University, involving Pat Hanrahan and Chris Stolte.**

Reference: Butz AM, *et al.*, *JAMA Pediatrics*, 2011.

Tableau’s Key Features

- **VizQL - A Visual Query Language that translates drag-and-drop actions into data queries and then expresses that data visually.**
  - **Read the paper: “Polaris: A System for Query, Analysis, and Visualization of Multidimensional Databases”.**



DOI:10.1145/1400214.1400234

Polaris: A System for Query, Analysis, and Visualization of Multidimensional Databases

By Chris Stolte, Diane Tang, and Pat Hanrahan

Abstract

During the last decade, multidimensional databases have become common in the business and scientific worlds. Analysis places significant demands on the interfaces to these databases. It must be possible for analysts to easily and incrementally change both the data and their views of it as they cycle between hypothesis and experimentation.

In this paper, we address these demands by presenting the Polaris formalism, a visual query language for precisely describing a wide range of table-based graphical presentations of data. This language compiles into both the queries and drawing commands necessary to generate the visualization, enabling us to design systems that closely integrate analysis and visualization. Using the Polaris formalism, we have built an interactive interface for exploring multidimensional databases that analysts can use to rapidly and incrementally build an expressive range of views of their data as they engage in a cycle of visual analysis.

1. INTRODUCTION

Nowadays, structured databases are widely used. Corporations store every sales transaction in large data warehouses. International research projects such as the Human Genome Project and Digital Sky Survey are generating massive scientific databases. Organizations such as the United Nations are making a wide range of global indicators on issues ranging from carbon emission to the adoption of technology publicly available via the Internet.

Unfortunately, our ability to collect and store data has rapidly exceeded our ability to analyze it. A major challenge in computer science is how to extract meaning from data: to discover structure, find patterns, and derive causal relationships. An analytical session cycles between hypothesis, experiment, and discovery. Often the path of exploration is unpredictable, and thus analysts need to be able to rapidly change both what data they are viewing and how they are viewing that data. This exploratory analysis process places significant demands on the human–computer interfaces to these databases. Few good tools exist.

In this paper, we present a formal approach to building visualization systems that addresses these demands. The first contribution is the Polaris formalism, a declarative visual query language that specifies a wide range of 2D graphic displays. The three key components of the formalism are (1) a table algebra that captures the structure of tables and spatial encodings, (2) a graphic taxonomy that results in an intuitive specification of graphic types, and (3) a system for effective visual encoding. This language allows for easily changing between different graphic displays as well as adding or removing data.

The second main contribution is the combination of this visual query language with the underlying database queries needed. This allows us to combine both visualization as well as the underlying data transformations to support the exploratory process.

The final contribution is the Polaris interface that allows users to incrementally construct a visual specification by dragging fields onto “shelves” (see Figure 1). Each intermediate specification is valid and corresponds to a graphical data display, giving the user quick visual feedback to support this analysis. This interface is built on top of the visual query language that specifies both the data and graphical transformations needed, thus combining statistical analysis and visualization. Polaris enables visual analysis by allowing an analyst to answer a question by composing a picture of what they want to see.

It has been 6 years since this work was originally published. In that time, the technology has been commercialized by Tableau Software as Tableau Desktop and is currently in use by thousands of companies and tens of thousands of users. As a result, we have gained considerable experience that has validated the effectiveness of the visual query language and interface and resulted in extensions and revisions to both.

2. OVERVIEW

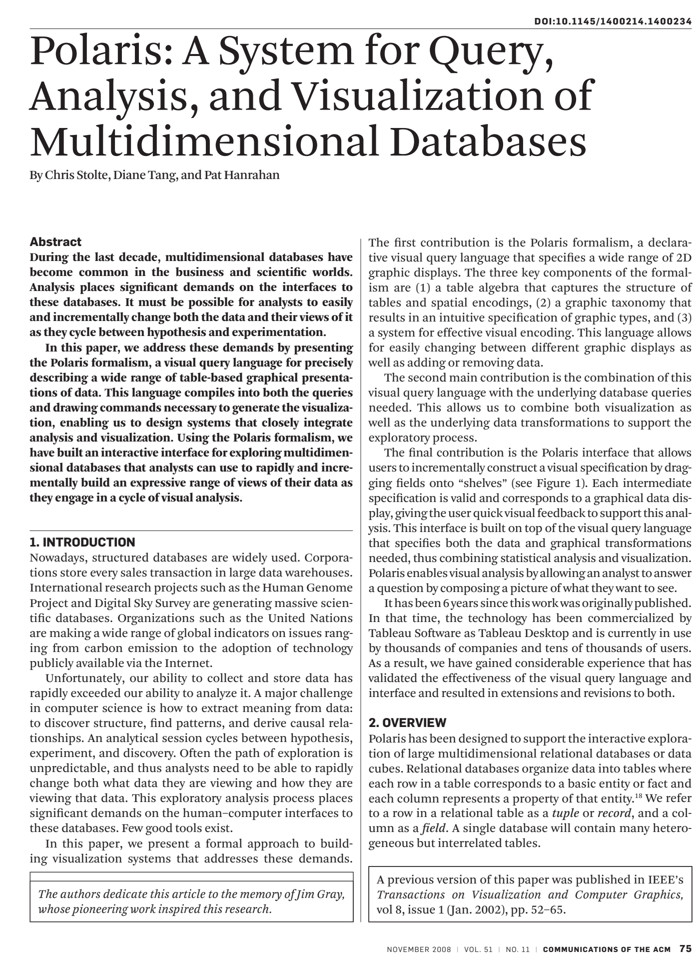
Polaris has been designed to support the interactive exploration of large multidimensional relational databases or data cubes. Relational databases organize data into tables where each row in a table corresponds to a basic entity or fact and each column represents a property of that entity.[18] We refer to a row in a relational table as a *tuple* or *record*, and a column as a *field*. A single database will contain many heterogeneous but interrelated tables.

The authors dedicate this article to the memory of Jim Gray, whose pioneering work inspired this research.

A previous version of this paper was published in IEEE’s *Transactions on Visualization and Computer Graphics,* vol 8, issue 1 (Jan. 2002), pp. 52–65.

Tableau's Key Features

- **VizQL - A Visual Query Language that translates drag-and-drop actions into data queries and then expresses that data visually.**
  - **Read the paper: "Polaris: A System for Query, Analysis, and Visualization of Multidimensional Databases".**



DOI:10.1145/1400214.1400234

Polaris: A System for Query, Analysis, and Visualization of Multidimensional Databases

By Chris Stolte, Diane Tang, and Pat Hanrahan

Abstract

During the last decade, multidimensional databases have become common in the business and scientific worlds. Analysis places significant demands on the interfaces to these databases. It must be possible for analysts to easily and incrementally change both the data and their views of it as they cycle between hypothesis and experimentation.

In this paper, we address these demands by presenting the Polaris formalism, a visual query language for precisely describing a wide range of table-based graphical presentations of data. This language compiles into both the queries and drawing commands necessary to generate the visualization, enabling us to design systems that closely integrate analysis and visualization. Using the Polaris formalism, we have built an interactive interface for exploring multidimensional databases that analysts can use to rapidly and incrementally build an expressive range of views of their data as they engage in a cycle of visual analysis.

1. INTRODUCTION

Nowadays, structured databases are widely used. Corporations store every sales transaction in large data warehouses. International research projects such as the Human Genome Project and Digital Sky Survey are generating massive scientific databases. Organizations such as the United Nations are making a wide range of global indicators on issues ranging from carbon emission to the adoption of technology publicly available via the Internet.

Unfortunately, our ability to collect and store data has rapidly exceeded our ability to analyze it. A major challenge in computer science is how to extract meaning from data: to discover structure, find patterns, and derive causal relationships. An analytical session cycles between hypothesis, experiment, and discovery. Often the path of exploration is unpredictable, and thus analysts need to be able to rapidly change both what data they are viewing and how they are viewing that data. This exploratory analysis process places significant demands on the human–computer interfaces to these databases. Few good tools exist.

In this paper, we present a formal approach to building visualization systems that addresses these demands.

The authors dedicate this article to the memory of Jim Gray, whose pioneering work inspired this research.

The first contribution is the Polaris formalism, a declarative visual query language that specifies a wide range of 2D graphic displays. The three key components of the formalism are (1) a table algebra that captures the structure of tables and spatial encodings, (2) a graphic taxonomy that results in an intuitive specification of graphic types, and (3) a system for effective visual encoding. This language allows for easily changing between different graphic displays as well as adding or removing data.

The second main contribution is the combination of this visual query language with the underlying database queries needed. This allows us to combine both visualization as well as the underlying data transformations to support the exploratory process.

The final contribution is the Polaris interface that allows users to incrementally construct a visual specification by dragging fields onto "shelves" (see Figure 1). Each intermediate specification is valid and corresponds to a graphical data display, giving the user quick visual feedback to support this analysis. This interface is built on top of the visual query language that specifies both the data and graphical transformations needed, thus combining statistical analysis and visualization. Polaris enables visual analysis by allowing an analyst to answer a question by composing a picture of what they want to see.

It has been 6 years since this work was originally published. In that time, the technology has been commercialized by Tableau Software as Tableau Desktop and is currently in use by thousands of companies and tens of thousands of users. As a result, we have gained considerable experience that has validated the effectiveness of the visual query language and interface and resulted in extensions and revisions to both.

2. OVERVIEW

Polaris has been designed to support the interactive exploration of large multidimensional relational databases or data cubes. Relational databases organize data into tables where each row in a table corresponds to a basic entity or fact and each column represents a property of that entity.[18] We refer to a row in a relational table as a *tuple* or *record*, and a column as a *field*. A single database will contain many heterogeneous but interrelated tables.

A previous version of this paper was published in IEEE's *Transactions on Visualization and Computer Graphics,* vol 8, issue 1 (Jan. 2002), pp. 52–65.

NOVEMBER 2008 | VOL. 51 | NO. 11 | COMMUNICATIONS OF THE ACM 75

Tableau’s Key Features

- **VizQL - A Visual Query Language that translates drag-and-drop actions into data queries and then expresses that data visually.**
  - **Read the paper: “Polaris: A System for Query, Analysis, and Visualization of Multidimensional Databases”.**

DOI:10.1145/1400214.1400234

Polaris: A System for Query, Analysis, and Visualization of Multidimensional Databases

By Chris Stolte, Diane Tang, and Pat Hanrahan

Abstract

During the last decade, multidimensional databases have become common in the business and scientific worlds. Analysis places significant demands on the interfaces to these databases. It must be possible for analysts to easily and incrementally change both the data and their views of it as they cycle between hypothesis and experimentation.

In this paper, we address these demands by presenting the Polaris formalism, a visual query language for precisely describing a wide range of table-based graphical presentations of data. This language compiles into both the queries and drawing commands necessary to generate the visualization, enabling us to design systems that closely integrate analysis and visualization. Using the Polaris formalism, we have built an interactive interface for exploring multidimensional databases that analysts can use to rapidly and incrementally build an expressive range of views of their data as they engage in a cycle of visual analysis.

1. INTRODUCTION

Nowadays, structured databases are widely used. Corporations store every sales transaction in large data warehouses. International research projects such as the Human Genome Project and Digital Sky Survey are generating massive scientific databases. Organizations such as the United Nations are making a wide range of global indicators on issues ranging from carbon emission to the adoption of technology publicly available via the Internet.

Unfortunately, our ability to collect and store data has rapidly exceeded our ability to analyze it. A major challenge in computer science is how to extract meaning from data: to discover structure, find patterns, and derive causal relationships. An analytical session cycles between hypothesis, experiment, and discovery. Often the path of exploration is unpredictable, and thus analysts need to be able to rapidly change both what data they are viewing and how they are viewing that data. This exploratory analysis process places significant demands on the human–computer interfaces to these databases. Few good tools exist.

In this paper, we present a formal approach to building visualization systems that addresses these demands. The first contribution is the Polaris formalism, a declarative visual query language that specifies a wide range of 2D graphic displays. The three key components of the formalism are (1) a table algebra that captures the structure of tables and spatial encodings, (2) a graphic taxonomy that results in an intuitive specification of graphic types, and (3) a system for effective visual encoding. This language allows for easily changing between different graphic displays as well as adding or removing data.

The second main contribution is the combination of this visual query language with the underlying database queries needed. This allows us to combine both visualization as well as the underlying data transformations to support the exploratory process.

The final contribution is the Polaris interface that allows users to incrementally construct a visual specification by dragging fields onto “shelves” (see Figure 1). Each intermediate specification is valid and corresponds to a graphical data display, giving the user quick visual feedback to support this analysis. This interface is built on top of the visual query language that specifies both the data and graphical transformations needed, thus combining statistical analysis and visualization. Polaris enables visual analysis by allowing an analyst to answer a question by composing a picture of what they want to see.

It has been 6 years since this work was originally published. In that time, the technology has been commercialized by Tableau Software as Tableau Desktop and is currently in use by thousands of companies and tens of thousands of users. As a result, we have gained considerable experience that has validated the effectiveness of the visual query language and interface and resulted in extensions and revisions to both.

2. OVERVIEW

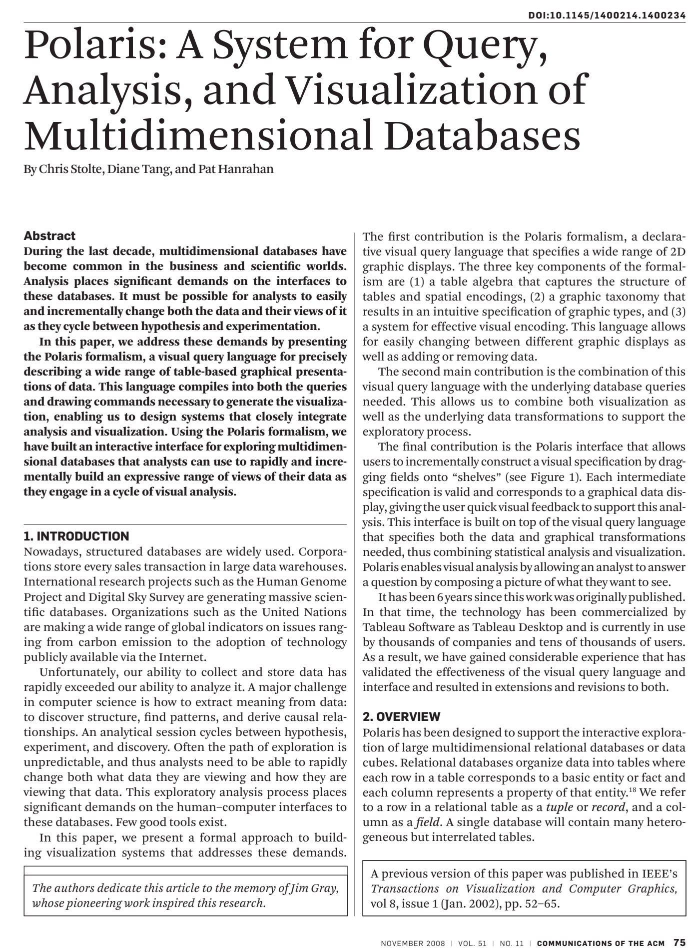
Polaris has been designed to support the interactive exploration of large multidimensional relational databases or data cubes. Relational databases organize data into tables where each row in a table corresponds to a basic entity or fact and each column represents a property of that entity.[18] We refer to a row in a relational table as a *tuple* or *record*, and a column as a *field*. A single database will contain many heterogeneous but interrelated tables.

The authors dedicate this article to the memory of Jim Gray, whose pioneering work inspired this research.

A previous version of this paper was published in IEEE’s *Transactions on Visualization and Computer Graphics*, vol 8, issue 1 (Jan. 2002), pp. 52–65.

Tableau's Key Features

- **VizQL - A Visual Query Language that translates drag-and-drop actions into data queries and then expresses that data visually.**
  - **Read the paper: "Polaris: A System for Query, Analysis, and Visualization of Multidimensional Databases".**


DOI:10.1145/1400214.1400234

Polaris: A System for Query, Analysis, and Visualization of Multidimensional Databases

By Chris Stolte, Diane Tang, and Pat Hanrahan

Abstract
During the last decade, multidimensional databases have become common in the business and scientific worlds. Analysis places significant demands on the interfaces to these databases. It must be possible for analysts to easily and incrementally change both the data and their views of it as they cycle between hypothesis and experimentation.

In this paper, we address these demands by presenting the Polaris formalism, a visual query language for precisely describing a wide range of table-based graphical presentations of data. This language compiles into both the queries and drawing commands necessary to generate the visualization, enabling us to design systems that closely integrate analysis and visualization. Using the Polaris formalism, we have built an interactive interface for exploring multidimensional databases that analysts can use to rapidly and incrementally build an expressive range of views of their data as they engage in a cycle of visual analysis.

1. INTRODUCTION
Nowadays, structured databases are widely used. Corporations store every sales transaction in large data warehouses. International research projects such as the Human Genome Project and Digital Sky Survey are generating massive scientific databases. Organizations such as the United Nations are making a wide range of global indicators on issues ranging from carbon emission to the adoption of technology publicly available via the Internet.

Unfortunately, our ability to collect and store data has rapidly exceeded our ability to analyze it. A major challenge in computer science is how to extract meaning from data: to discover structure, find patterns, and derive causal relationships. An analytical session cycles between hypothesis, experiment, and discovery. Often the path of exploration is unpredictable, and thus analysts need to be able to rapidly change both what data they are viewing and how they are viewing that data. This exploratory analysis process places significant demands on the human–computer interfaces to these databases. Few good tools exist.

In this paper, we present a formal approach to building visualization systems that addresses these demands. The first contribution is the Polaris formalism, a declarative visual query language that specifies a wide range of 2D graphic displays. The three key components of the formalism are (1) a table algebra that captures the structure of tables and spatial encodings, (2) a graphic taxonomy that results in an intuitive specification of graphic types, and (3) a system for effective visual encoding. This language allows for easily changing between different graphic displays as well as adding or removing data.

The second main contribution is the combination of this visual query language with the underlying database queries needed. This allows us to combine both visualization as well as the underlying data transformations to support the exploratory process.

The final contribution is the Polaris interface that allows users to incrementally construct a visual specification by dragging fields onto "shelves" (see Figure 1). Each intermediate specification is valid and corresponds to a graphical data display, giving the user quick visual feedback to support this analysis. This interface is built on top of the visual query language that specifies both the data and graphical transformations needed, thus combining statistical analysis and visualization. Polaris enables visual analysis by allowing an analyst to answer a question by composing a picture of what they want to see.

It has been 6 years since this work was originally published. In that time, the technology has been commercialized by Tableau Software as Tableau Desktop and is currently in use by thousands of companies and tens of thousands of users. As a result, we have gained considerable experience that has validated the effectiveness of the visual query language and interface and resulted in extensions and revisions to both.

2. OVERVIEW
Polaris has been designed to support the interactive exploration of large multidimensional relational databases or data cubes. Relational databases organize data into tables where each row in a table corresponds to a basic entity or fact and each column represents a property of that entity.[18] We refer to a row in a relational table as a *tuple* or *record*, and a column as a *field*. A single database will contain many heterogeneous but interrelated tables.

The authors dedicate this article to the memory of Jim Gray, whose pioneering work inspired this research.

A previous version of this paper was published in IEEE's *Transactions on Visualization and Computer Graphics*, vol 8, issue 1 (Jan. 2002), pp. 52–65.

NOVEMBER 2008 | VOL. 51 | NO. 11 | COMMUNICATIONS OF THE ACM 75

DOI:10.1145/1400214.1400234

Polaris: A System for Query, Analysis, and Visualization of Multidimensional Databases

By Chris Stolte, Diane Tang, and Pat Hanrahan

Abstract

During the last decade, multidimensional databases have become common in the business and scientific worlds. Analysis places significant demands on the interfaces to these databases. It must be possible for analysts to easily and incrementally change both the data and their views of it as they cycle between hypothesis and experimentation.

In this paper, we address these demands by presenting the Polaris formalism, a visual query language for precisely describing a wide range of table-based graphical presentations of data. This language compiles into both the queries and drawing commands necessary to generate the visualization, enabling us to design systems that closely integrate analysis and visualization. Using the Polaris formalism, we have built an interactive interface for exploring multidimensional databases that analysts can use to rapidly and incrementally build an expressive range of views of their data as they engage in a cycle of visual analysis.

1. INTRODUCTION

Nowadays, structured databases are widely used. Corporations store every sales transaction in large data warehouses. International research projects such as the Human Genome Project and Digital Sky Survey are generating massive scientific databases. Organizations such as the United Nations are making a wide range of global indicators on issues rang-

The first contribution is the Polaris formalism, a declarative visual query language that specifies a wide range of 2D graphic displays. The three key components of the formalism are (1) a table algebra that captures the structure of tables and spatial encodings, (2) a graphic taxonomy that results in an intuitive specification of graphic types, and (3) a system for effective visual encoding. This language allows for easily changing between different graphic displays as well as adding or removing data.

The second main contribution is the combination of this visual query language with the underlying database queries needed. This allows us to combine both visualization as well as the underlying data transformations to support the exploratory process.

The final contribution is the Polaris interface that allows users to incrementally construct a visual specification by dragging fields onto "shelves" (see Figure 1). Each intermediate specification is valid and corresponds to a graphical data display, giving the user quick visual feedback to support this analysis. This interface is built on top of the visual query language that specifies both the data and graphical transformations needed, thus combining statistical analysis and visualization. Polaris enables visual analysis by allowing an analyst to answer a question by composing a picture of what they want to see.

It has been 6 years since this work was originally published. In that time, the technology has been commercialized by Tableau Software as Tableau Desktop and is currently in use

Abstract

During the last decade, multidimensional databases have become common in the business and scientific worlds. Analysis places significant demands on the interfaces to these databases. It must be possible for analysts to easily and incrementally change both the data and their views of it as they cycle between hypothesis and experimentation.

In this paper, we address these demands by presenting the Polaris formalism, a visual query language for precisely describing a wide range of table-based graphical presentations of data. This language compiles into both the queries and drawing commands necessary to generate the visualization, enabling us to design systems that closely integrate analysis and visualization. Using the Polaris formalism, we have built an interactive interface for exploring multidimensional databases that analysts can use to rapidly and incrementally build an expressive range of views of their data as they engage in a cycle of visual analysis.

Abstract

During the last decade, multidimensional databases have become common in the business and scientific worlds. Analysis places significant demands on the interfaces to these databases. It must be possible for analysts to easily and incrementally change both the data and their views of it as they cycle between hypothesis and experimentation.

In this paper, we address these demands by presenting the Polaris formalism, a visual query language for precisely describing a wide range of table-based graphical presentations of data. This language compiles into both the queries and drawing commands necessary to generate the visualization, enabling us to design systems that closely integrate analysis and visualization. Using the Polaris formalism, we have built an interactive interface for exploring multidimensional databases that analysts can use to rapidly and incrementally build an expressive range of views of their data as they engage in a cycle of visual analysis.

Abstract

During the last decade, multidimensional databases have become common in the business and scientific worlds. Analysis places significant demands on the interfaces to these databases. It must be possible for analysts to easily and incrementally change both the data and their views of it as they cycle between hypothesis and experimentation.

In this paper, we address these demands by presenting the Polaris formalism, a visual query language for precisely describing a wide range of table-based graphical presentations of data. This language compiles into both the queries and drawing commands necessary to generate the visualization, enabling us to design systems that closely integrate analysis and visualization. Using the Polaris formalism, we have built an interactive interface for exploring multidimensional databases that analysts can use to rapidly and incrementally build an expressive range of views of their data as they engage in a cycle of visual analysis.

Abstract

During the last decade, multidimensional databases have become common in the business and scientific worlds. Analysis places significant demands on the interfaces to these databases. It must be possible for analysts to easily and incrementally change both the data and their views of it as they cycle between hypothesis and experimentation.

In this paper, we address these demands by presenting the Polaris formalism, a visual query language for precisely describing a wide range of table-based graphical presentations of data. This language compiles into both the queries and drawing commands necessary to generate the visualization, enabling us to design systems that closely integrate analysis and visualization. Using the Polaris formalism, we have built an interactive interface for exploring multidimensional databases that analysts can use to rapidly and incrementally build an expressive range of views of their data as they engage in a cycle of visual analysis.

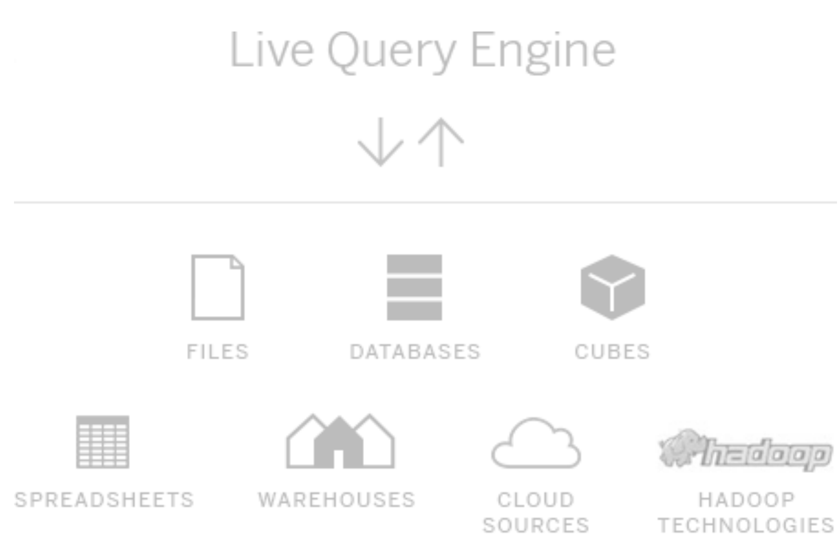
Abstract

During the last decade, multidimensional databases have become common in the business and scientific worlds. Analysis places significant demands on the interfaces to these databases. It must be possible for analysts to easily and incrementally change both the data and their views of it as they cycle between hypothesis and experimentation.

In this paper, we address these demands by presenting the Polaris formalism, a visual query language for precisely describing a wide range of table-based graphical presentations of data. This language compiles into both the queries and drawing commands necessary to generate the visualization, enabling us to design systems that closely integrate analysis and visualization. Using the Polaris formalism, we have built an interactive interface for exploring multidimensional databases that analysts can use to rapidly and incrementally build an expressive range of views of their data as they engage in a cycle of visual analysis.

Tableau's Key Features

- **Live Query Engine - A technology that lets people query databases, cubes, warehouses, cloud sources, spreadsheets, etc. without any programming knowledge.**
  - **Heterogeneous data sources can be combined and data made available in a transparent way.**



- **With In-Memory Data Engine - uses the complete memory hierarchy (Disk-RAM-L1 Cache) on ordinary computers to speedup access to slow databases.**

Tableau Family of Products

Tableau Family of Products

- **Tableau Desktop - Runs on a desktop computer, can connect to remote databases and services.**

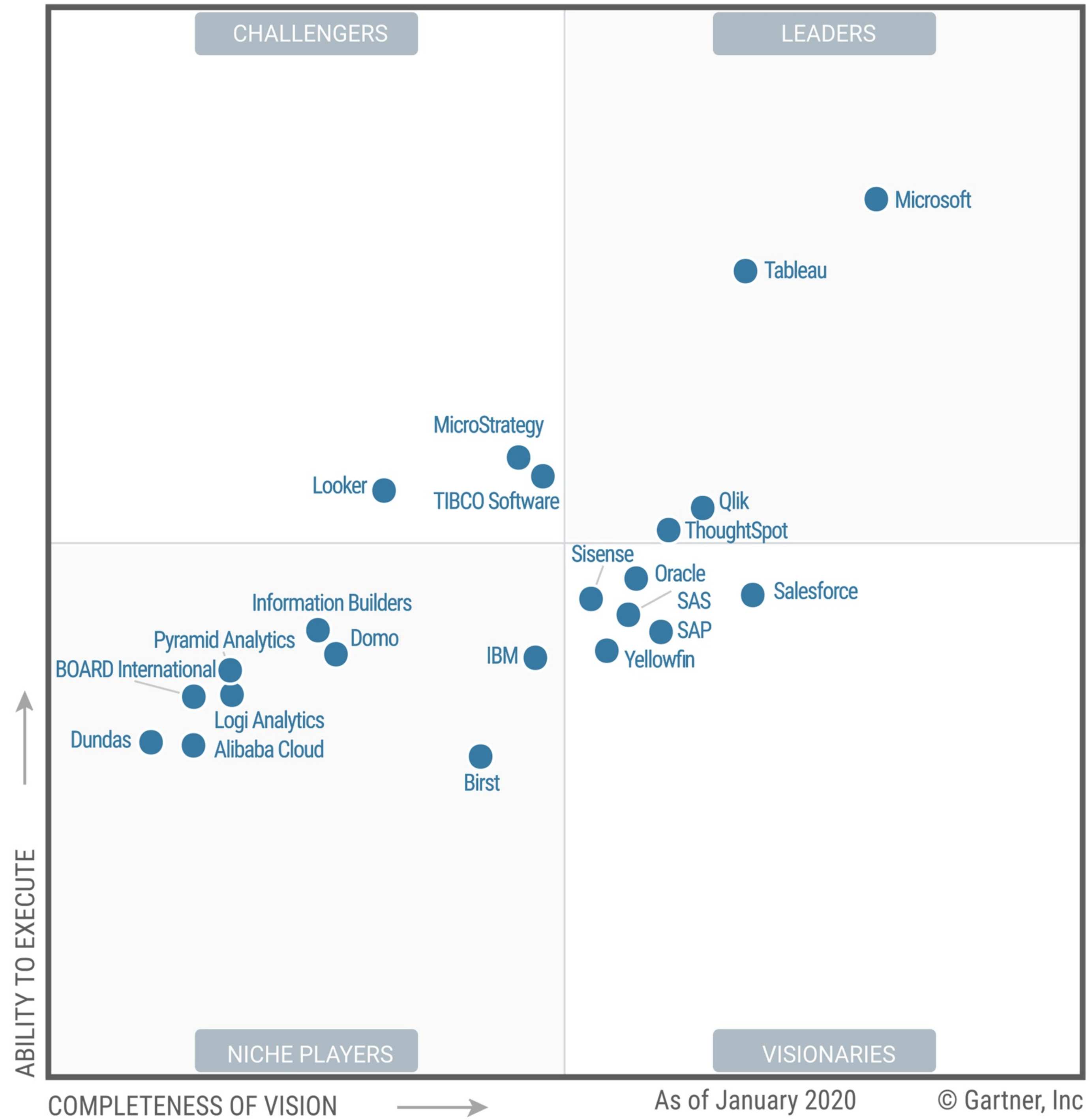
Tableau Family of Products

- **Tableau Desktop - Runs on a desktop computer, can connect to remote databases and services.**
- **Tableau Prep - To combine, shape, and clean the data**

Tableau Family of Products

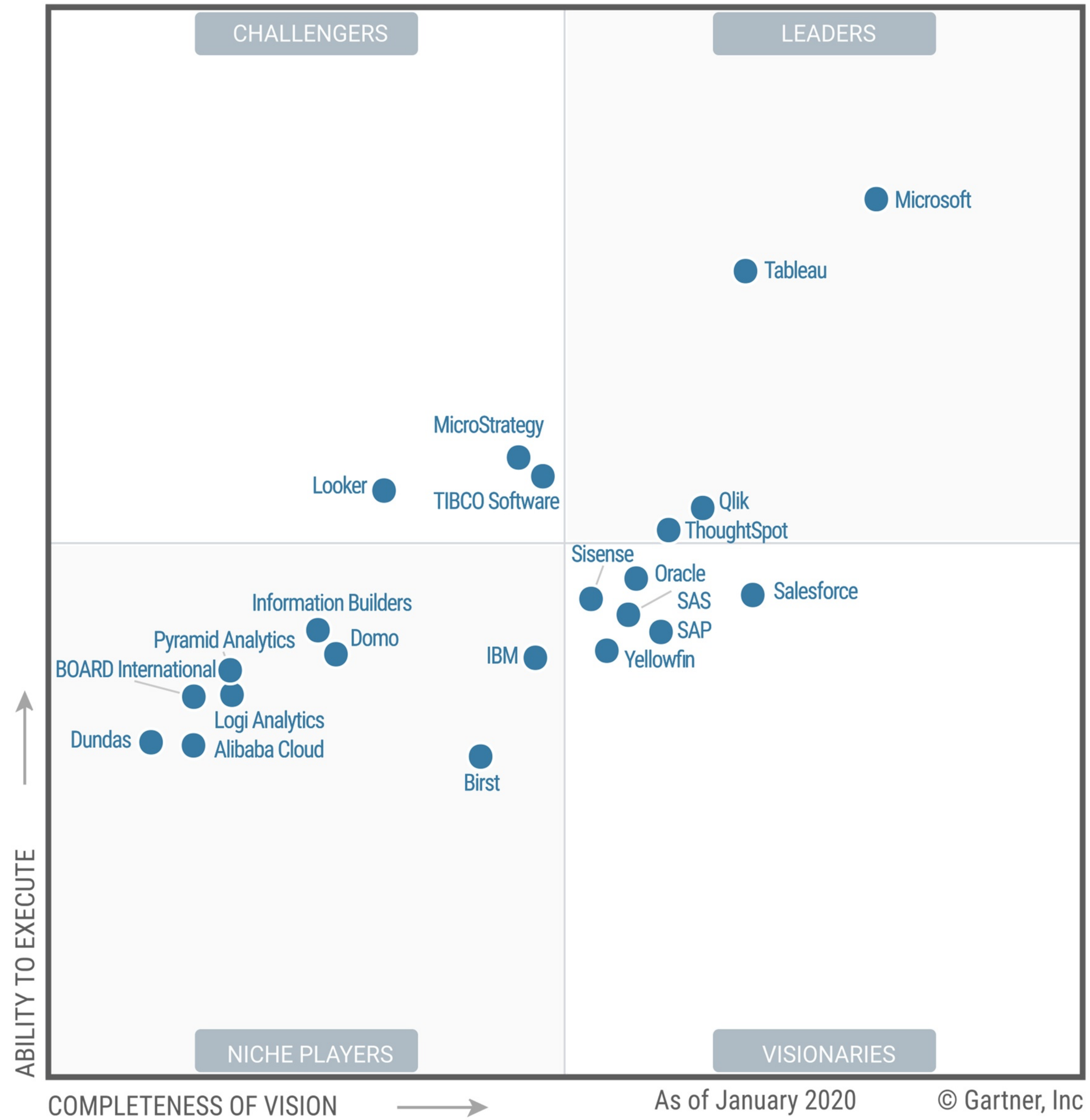
- Tableau Desktop - Runs on a desktop computer, can connect to remote databases and services.
- Tableau Prep - To combine, shape, and clean the data
- Tableau Server - Allows to publish and share your data whether in-house or in the cloud and collaborate with teams.
- Tableau Online - Hosted (cloud based) version of Tableau Server.
- Tableau Public - A simpler version of Tableau Desktop with cloud based storage/profile to store/share visualizations publicly.

Tableau in the Gartner Magic Quadrant



Source: Gartner (February 2020)

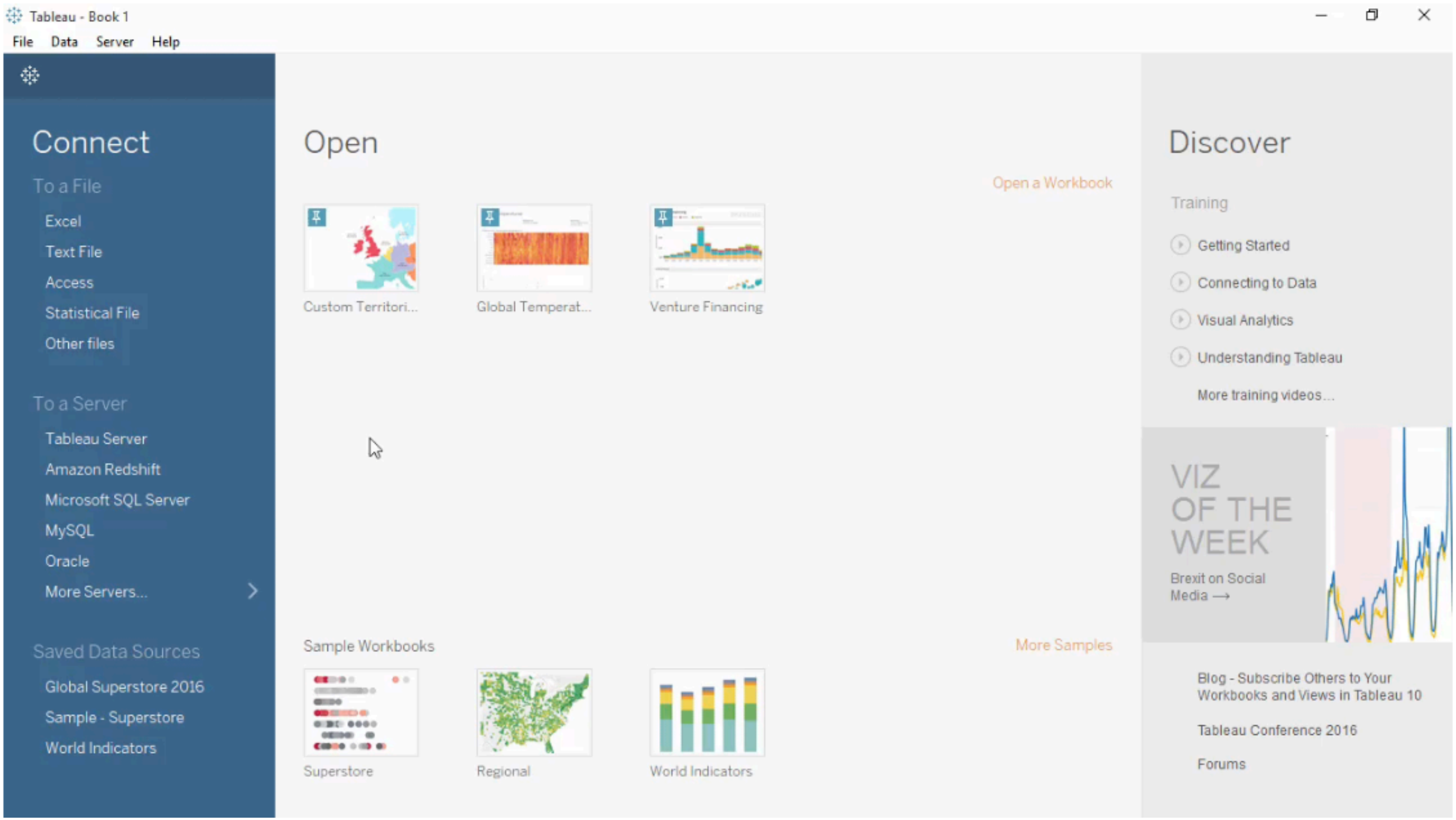
Tableau in the Gartner Magic Quadrant



Pricing

Source: Gartner (February 2020)

Tableau Software Overview



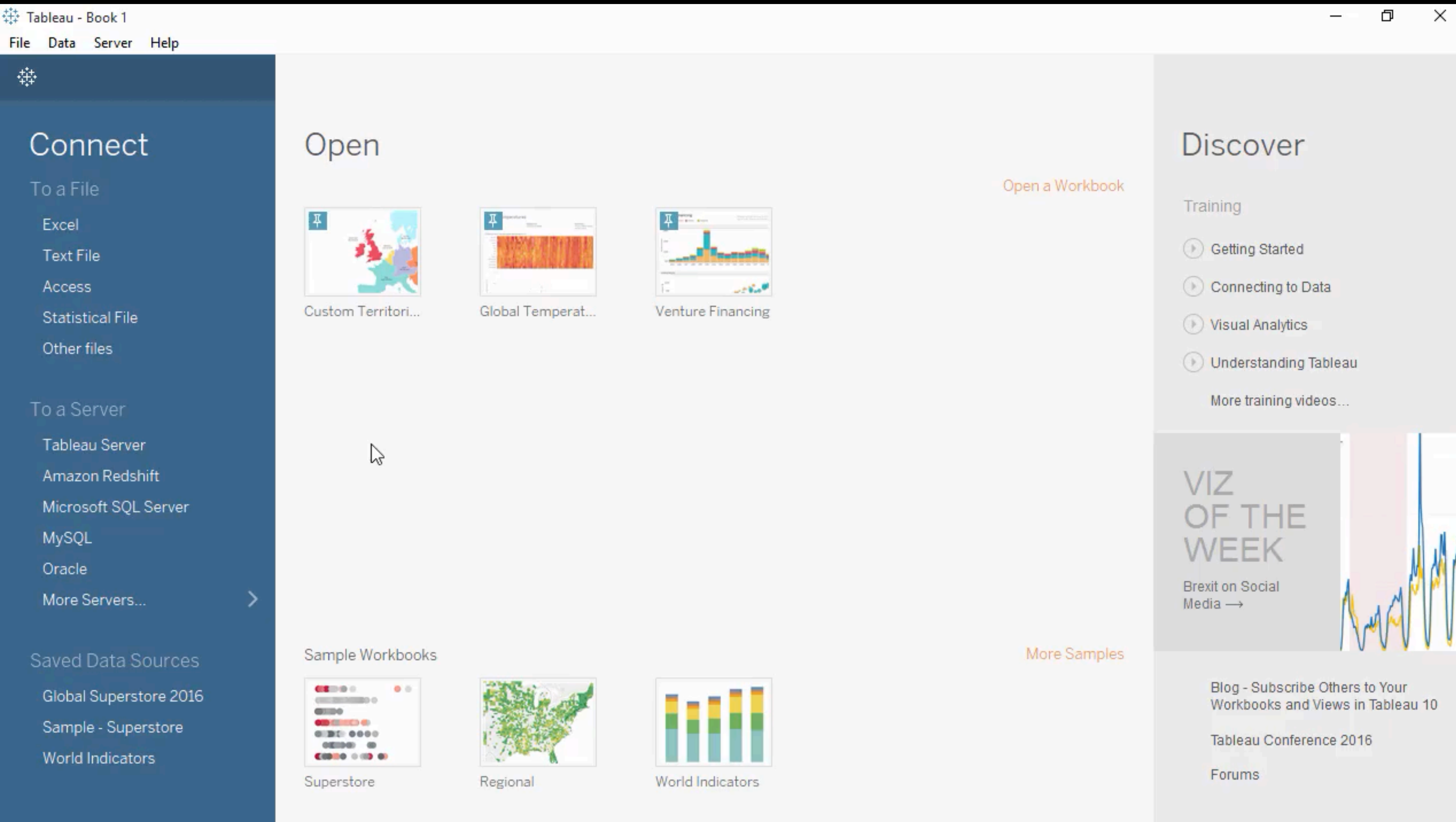


Tableau - Book 1
File
Data
Server
Help
Connect
To a File
Excel
Text File
Access
Statistical File
Other files
To a Server
Tableau Server
Amazon Redshift
Microsoft SQL Server
MySQL
Oracle
More Servers...
Saved Data Sources
Global Superstore 2016
Sample - Superstore
World Indicators
Open
Open a Workbook
Custom Territori...
Global Temperat...
Venture Financing
Sample Workbooks
More Samples
Superstore
Regional
World Indicators
Discover
Training
Getting Started
Connecting to Data
Visual Analytics
Understanding Tableau
More training videos...
VIZ OF THE WEEK
Brexit on Social Media ⟶
Blog - Subscribe Others to Your Workbooks and Views in Tableau 10
Tableau Conference 2016
Forums

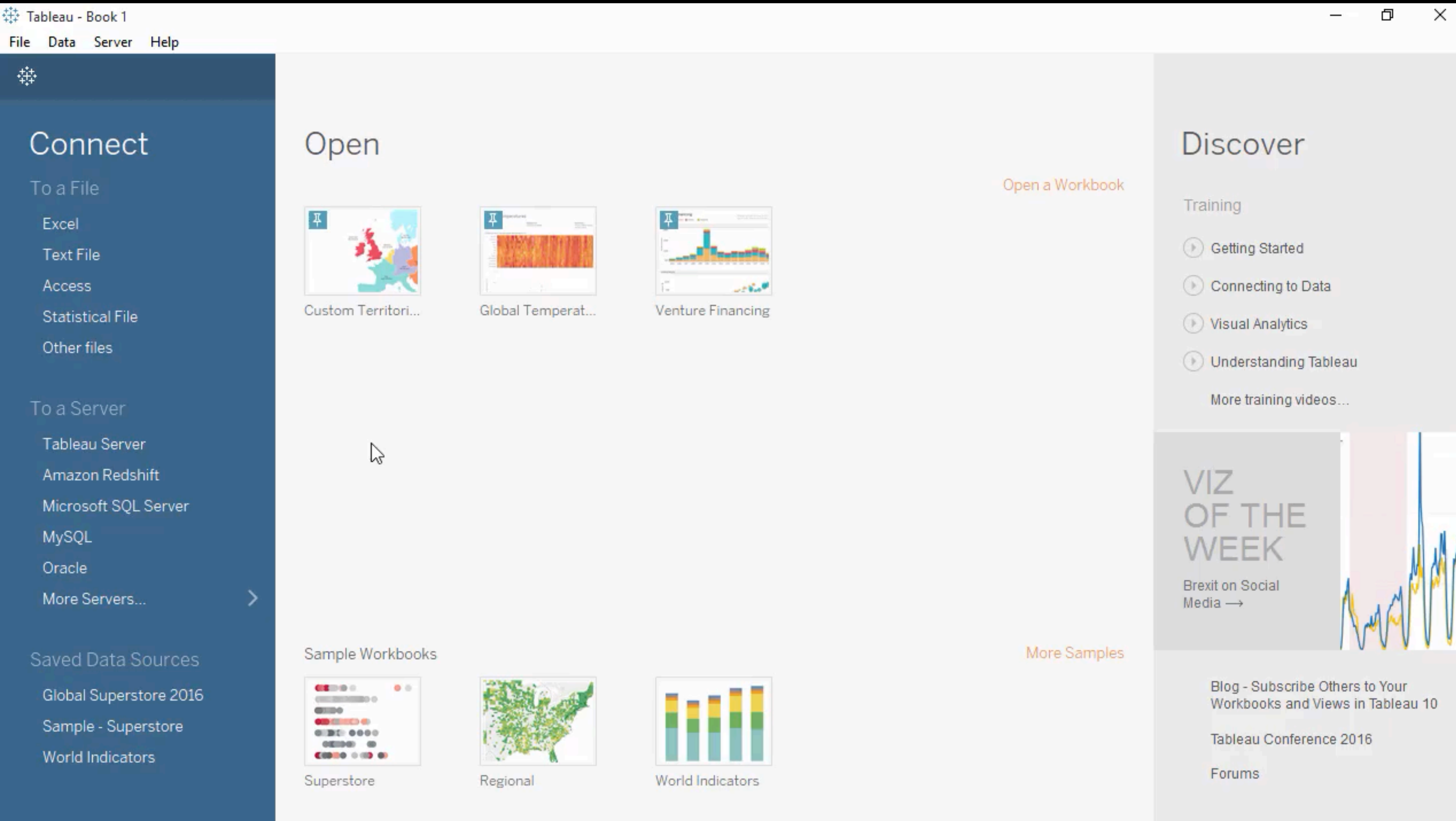


Tableau - Book 1
File
Data
Server
Help
Connect
To a File
Excel
Text File
Access
Statistical File
Other files
To a Server
Tableau Server
Amazon Redshift
Microsoft SQL Server
MySQL
Oracle
More Servers...
Saved Data Sources
Global Superstore 2016
Sample - Superstore
World Indicators
Open
Open a Workbook
Custom Territori...
Global Temperat...
Venture Financing
Sample Workbooks
More Samples
Superstore
Regional
World Indicators
Discover
Training
Getting Started
Connecting to Data
Visual Analytics
Understanding Tableau
More training videos...
VIZ OF THE WEEK
Brexit on Social Media ⟶
Blog - Subscribe Others to Your Workbooks and Views in Tableau 10
Tableau Conference 2016
Forums

Connecting to Data

To Files

- ✓ Other Files (*.twb *.twbx *.tbm *.tds *.tdsx *.tde *.hyper *.xls *.xlsx *.x...s7bdat *.sav *.rda *.rdata *.json *.pdf *.kml *.mif *.shp *.geojson *.zip)
- Tableau Workbooks (*.twb)
- Tableau Packaged Workbooks (*.twbx)
- Tableau Data Sources (*.tds)
- Tableau Packaged Data Sources (*.tdsx)
- Tableau Data Extracts (*.hyper *.tde)
- Excel Workbooks (*.xls *.xlsx *.xlsm)
- Character Delimited Files (*.csv)
- Text Files (*.txt)
- Adobe Portable Document Format (*.pdf)
- Statistical Files (*.sav *.sas7bdat *.rda *.rdata)
- Tab Delimited Files (*.tab *.tsv)
- Spatial Files (*.kml *.shp *.tab *.mif *.geojson *.zip *.json *.topojson)
- JSON Files (*.json)
- All Files (*.*)

Connecting to Data

- **Servers**

Tableau Server
Amazon Athena
Amazon Aurora
Amazon EMR Hadoop Hive
Amazon Redshift
Anaplan
Apache Drill
Aster Database
Azure SQL Data Warehouse
Box
Cloudera Hadoop
Denodo
Dropbox
Exasol
Firebird
Google Ads
Google Analytics
Google BigQuery
Google Cloud SQL

Google Drive
Google Sheets
Hortonworks Hadoop Hive
Intuit QuickBooks Online
Intuit QuickBooks Online (9.3-2018.1)
Kognitio
MapR Hadoop Hive
MariaDB
Marketo
MemSQL
Microsoft SQL Server
MongoDB BI Connector
MySQL
OData
OneDrive
Oracle
Oracle Eloqua
Pivotal Greenplum Database
PostgreSQL

Presto
Salesforce
SAP HANA
ServiceNow ITSM
SharePoint Lists
Snowflake
Spark SQL
Teradata
Vertica
Web Data Connector

Other Databases (JDBC)
Other Databases (ODBC)

Connecting to Data

- **Integrated Data from multiple data sources**
- **Joins, including spatial join**
- **Rename fields, Change the Data Type, Compute new columns**
- **Connection Live versus Extract**

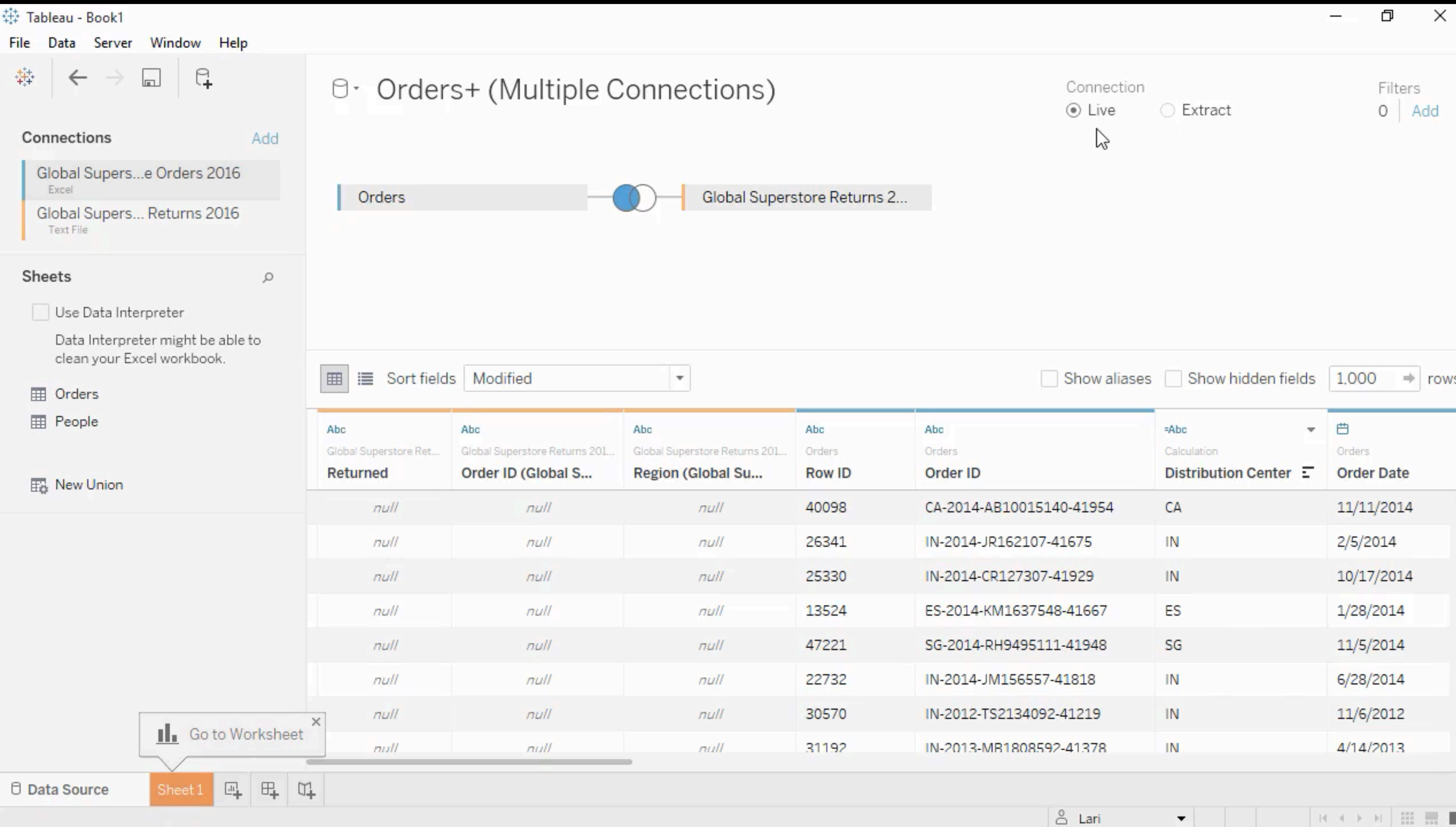


Tableau - Book1
File
Data
Server
Window
Help
Orders+ (Multiple Connections)
Connection
Live
Extract
Filters
0
Add
Connections
Add
Global Supers...e Orders 2016
Excel
Global Supers... Returns 2016
Text File
Orders
Global Superstore Returns 2...
Sheets
Use Data Interpreter
Data Interpreter might be able to clean your Excel workbook.
Orders
People
New Union
Sort fields
Modified
Show aliases
Show hidden fields
1,000
rows
Global Superstore Ret...
Returned
Global Superstore Returns 201...
Order ID (Global S...
Global Superstore Returns 201...
Region (Global Su...
Orders
Row ID
Orders
Order ID
Calculation
Distribution Center
Orders
Order Date
null null null 40098 CA-2014-AB10015140-41954 CA 11/11/2014
null null null 26341 IN-2014-JR162107-41675 IN 2/5/2014
null null null 25330 IN-2014-CR127307-41929 IN 10/17/2014
null null null 13524 ES-2014-KM1637548-41667 ES 1/28/2014
null null null 47221 SG-2014-RH9495111-41948 SG 11/5/2014
null null null 22732 IN-2014-JM156557-41818 IN 6/28/2014
null null null 30570 IN-2012-TS2134092-41219 IN 11/6/2012
null null null 31192 IN-2013-MB1808592-41378 IN 4/14/2013
Go to Worksheet
Data Source
Sheet1
Lari

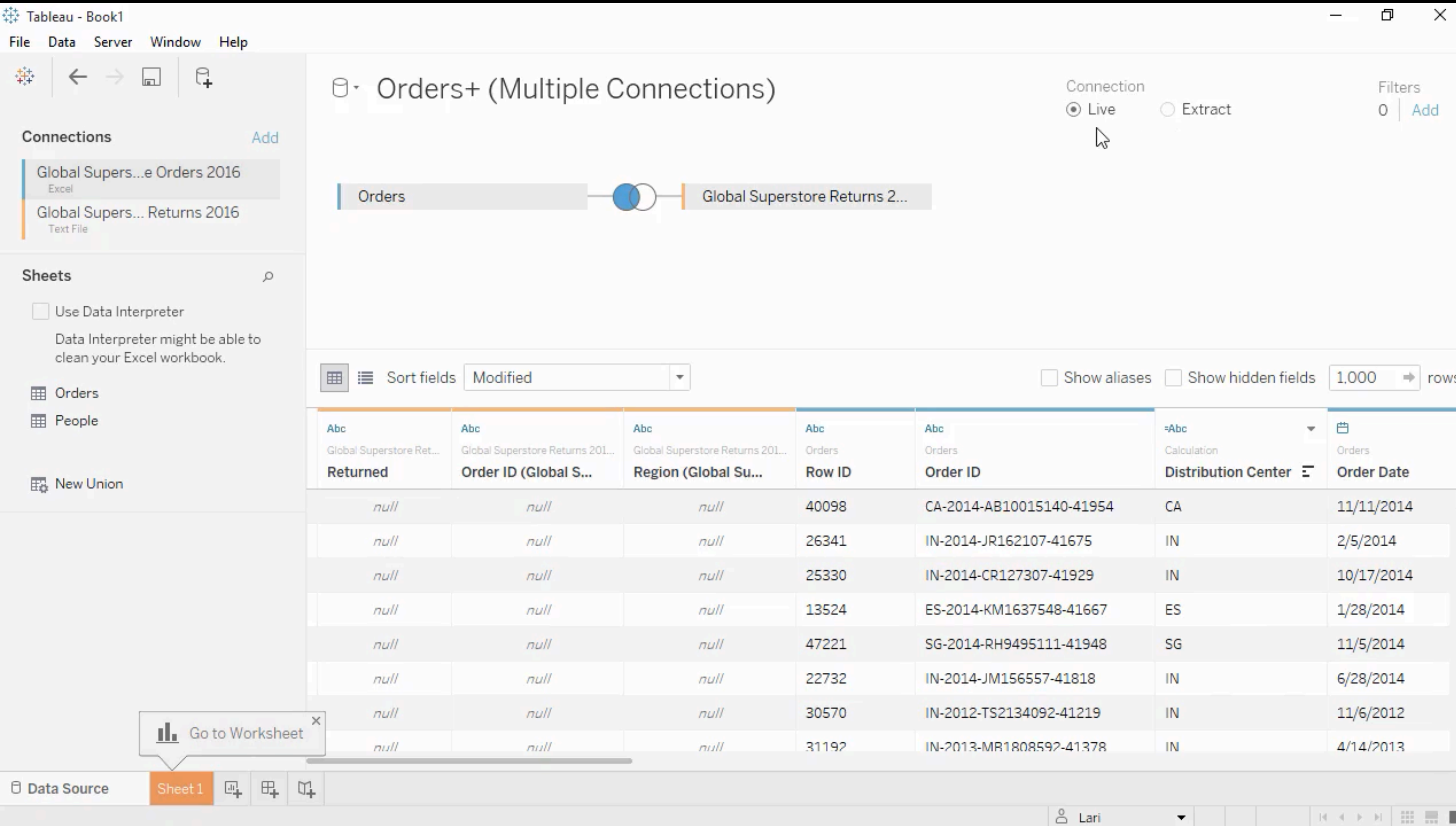


Tableau - Book1
File Data Server Window Help
Orders+ (Multiple Connections)
Connection
Live
Extract
Filters
0
Add
Connections
Add
Global Supers...e Orders 2016
Excel
Global Supers... Returns 2016
Text File
Orders
Global Superstore Returns 2...
Sheets
Use Data Interpreter
Data Interpreter might be able to clean your Excel workbook.
Orders
People
New Union
Sort fields
Modified
Show aliases
Show hidden fields
1,000
rows
Returned
Order ID (Global S...
Region (Global Su...
Row ID
Order ID
Distribution Center
Order Date
null null null 40098 CA-2014-AB10015140-41954 CA 11/11/2014
null null null 26341 IN-2014-JR162107-41675 IN 2/5/2014
null null null 25330 IN-2014-CR127307-41929 IN 10/17/2014
null null null 13524 ES-2014-KM1637548-41667 ES 1/28/2014
null null null 47221 SG-2014-RH9495111-41948 SG 11/5/2014
null null null 22732 IN-2014-JM156557-41818 IN 6/28/2014
null null null 30570 IN-2012-TS2134092-41219 IN 11/6/2012
Go to Worksheet
Data Source
Sheet1
Lari

First Graphic

- **Drag and Drop**
- **Dimension and Measures**
- **Columns and Rows**
- **Discrete versus continuous**
- **Query the data before building the graphic**

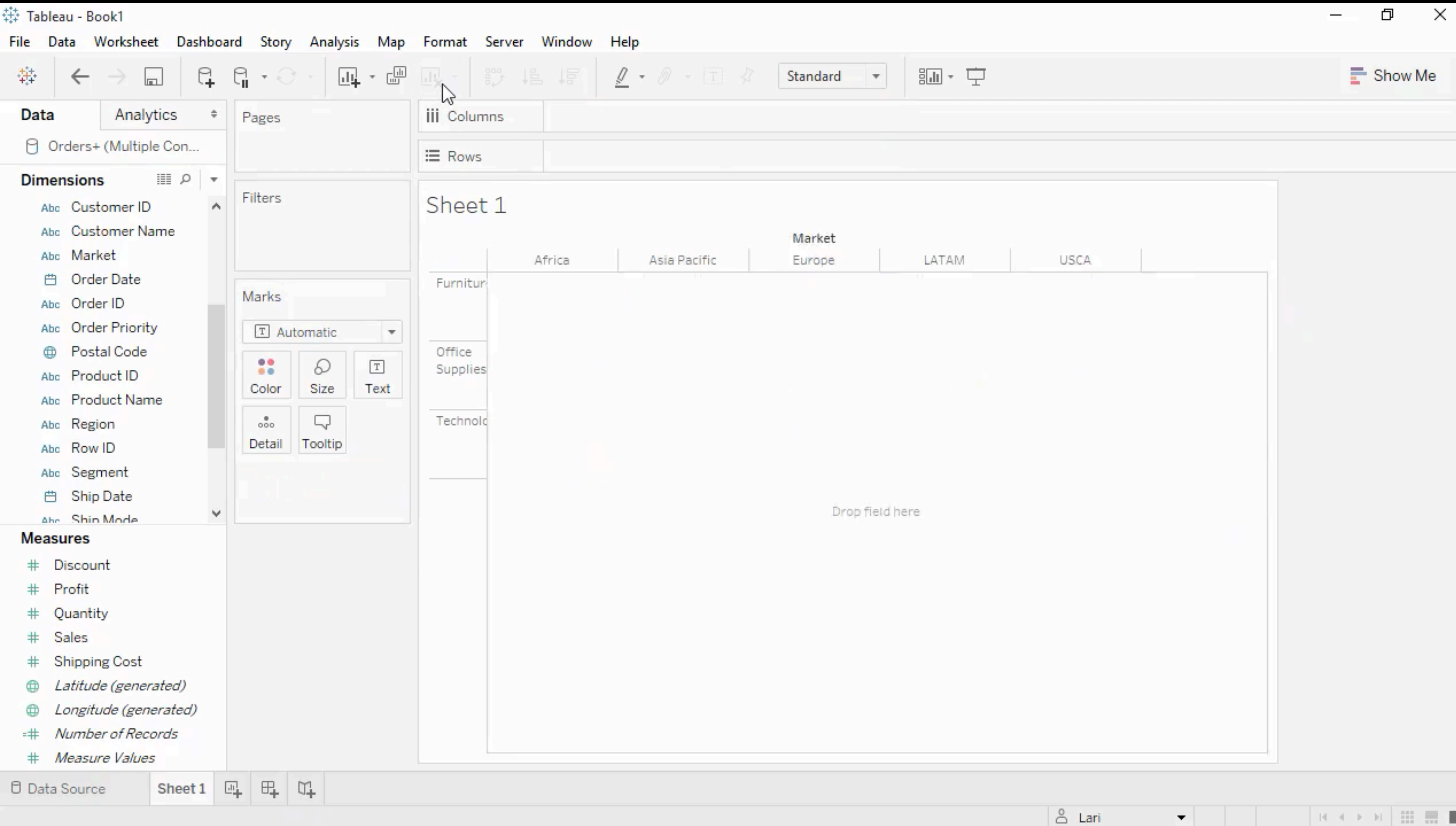


Tableau - Book1
File
Data
Worksheet
Dashboard
Story
Analysis
Map
Format
Server
Window
Help
Standard
Show Me
Data
Analytics
Orders+ (Multiple Con...
Pages
Columns
Rows
Filters
Marks
Automatic
Color
Size
Text
Detail
Tooltip
Dimensions
Customer ID
Customer Name
Market
Order Date
Order ID
Order Priority
Postal Code
Product ID
Product Name
Region
Row ID
Segment
Ship Date
Ship Mode
Measures
Discount
Profit
Quantity
Sales
Shipping Cost
Latitude (generated)
Longitude (generated)
Number of Records
Measure Values
Sheet 1
Market
Africa
Asia Pacific
Europe
LATAM
USCA
Furnitur
Office Supplies
Technolo
Drop field here
Data Source
Sheet 1
Lari

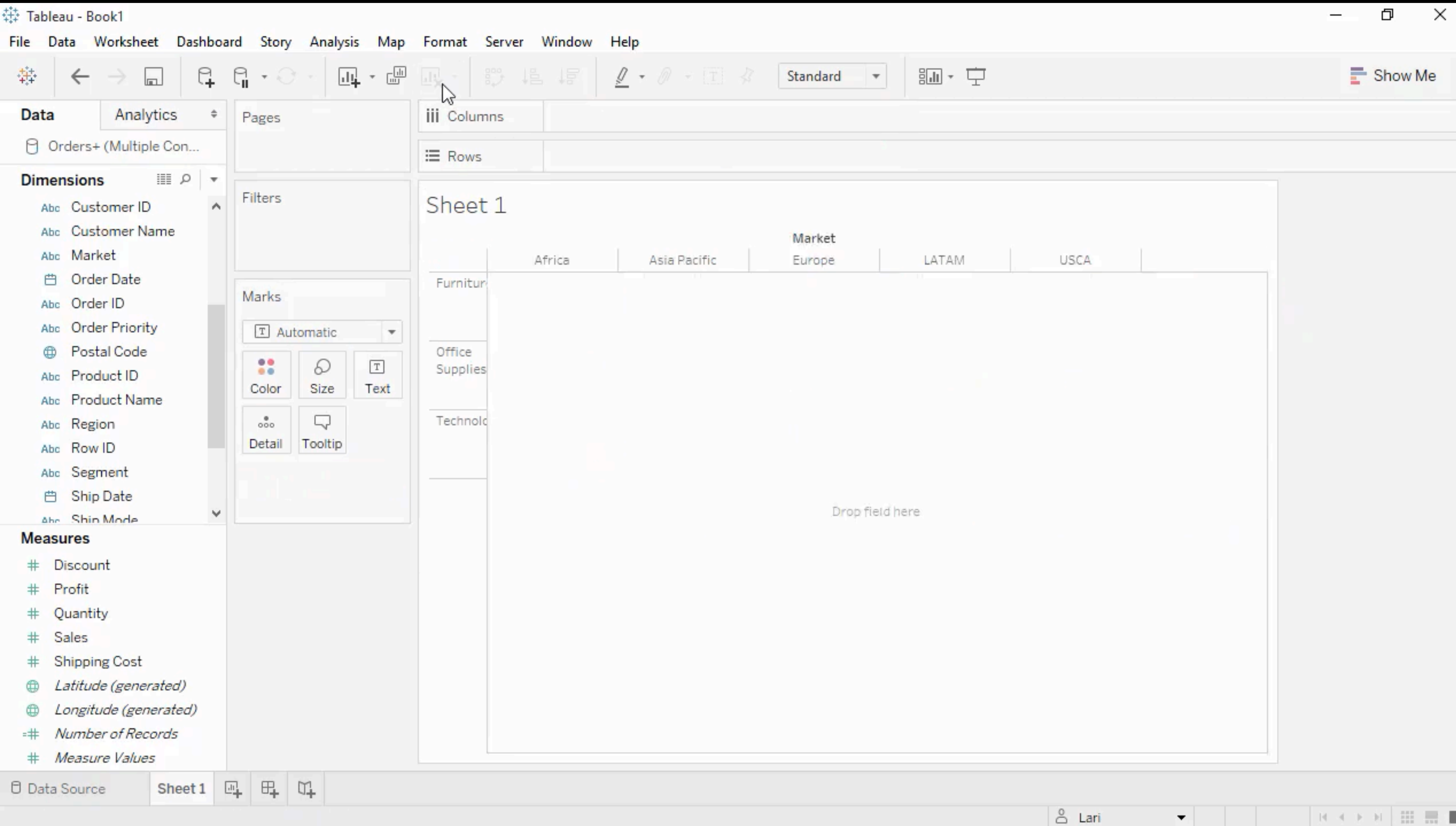


Tableau - Book1
File Data Worksheet Dashboard Story Analysis Map Format Server Window Help
Standard
Show Me
Data
Analytics
Orders+ (Multiple Con...
Pages
Columns
Rows
Filters
Marks
Automatic
Color
Size
Text
Detail
Tooltip
Dimensions
Customer ID
Customer Name
Market
Order Date
Order ID
Order Priority
Postal Code
Product ID
Product Name
Region
Row ID
Segment
Ship Date
Measures
Discount
Profit
Quantity
Sales
Shipping Cost
Latitude (generated)
Longitude (generated)
Number of Records
Measure Values
Sheet 1
Market
Africa
Asia Pacific
Europe
LATAM
USCA
Office Supplies
Drop field here
Data Source
Sheet 1
Lari

Drill Down

- **Query the data before building the graphic**

- **Drill down**

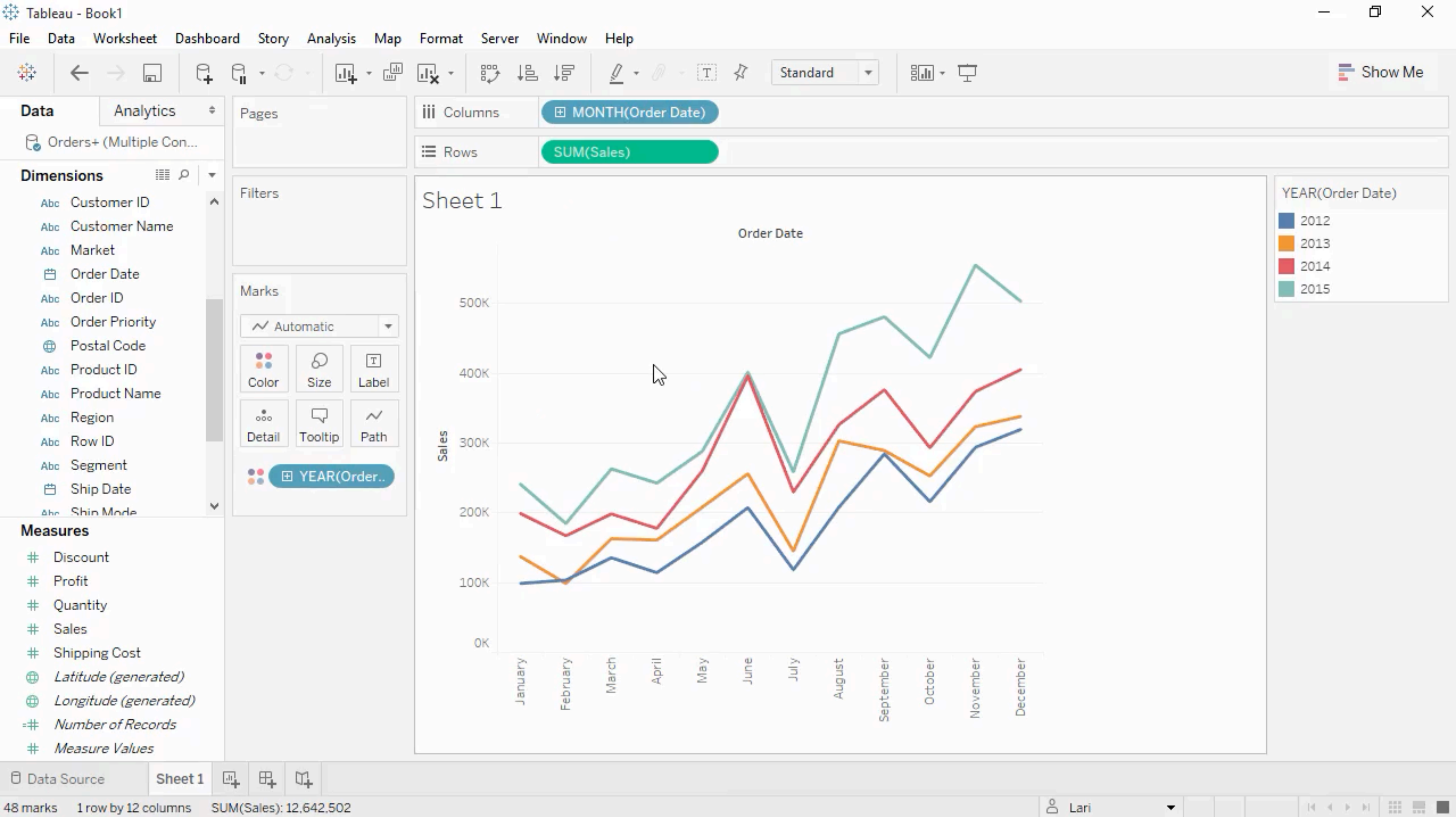


Tableau - Book1
File Data Worksheet Dashboard Story Analysis Map Format Server Window Help
Standard
Show Me
Data Analytics
Orders+ (Multiple Con...
Dimensions
Customer ID
Customer Name
Market
Order Date
Order ID
Order Priority
Postal Code
Product ID
Product Name
Region
Row ID
Segment
Ship Date
Ship Mode
Measures
Discount
Profit
Quantity
Sales
Shipping Cost
Latitude (generated)
Longitude (generated)
Number of Records
Measure Values
Pages
Filters
Marks
Automatic
Color Size Label
Detail Tooltip Path
YEAR(Order..
Columns MONTH(Order Date)
Rows SUM(Sales)
Sheet 1
Order Date
Sales
500K
400K
300K
200K
100K
0K
January February March April May June July August September October November December
YEAR(Order Date)
2012
2013
2014
2015
Data Source Sheet 1
48 marks 1 row by 12 columns SUM(Sales): 12,642,502
Lari

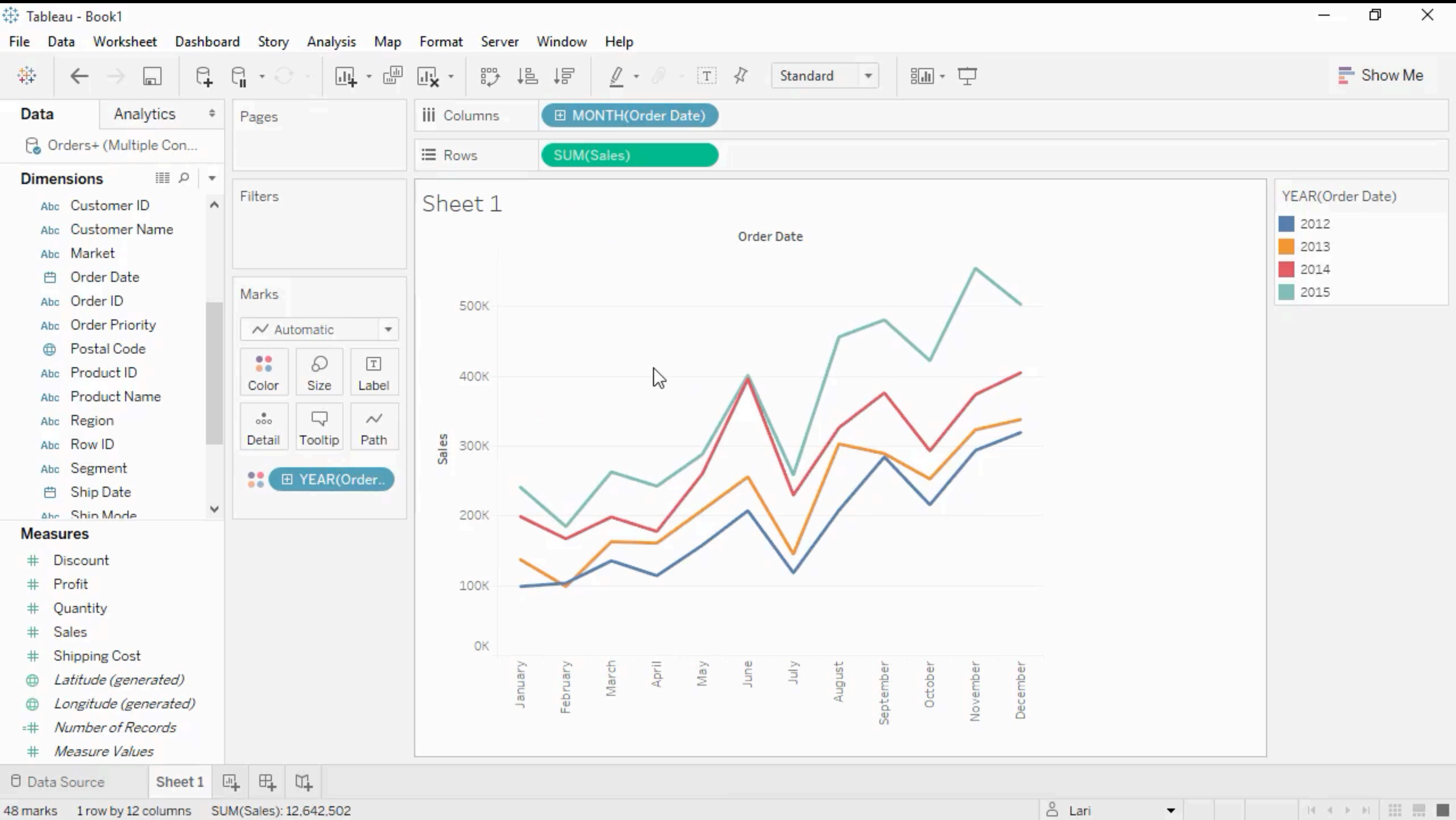


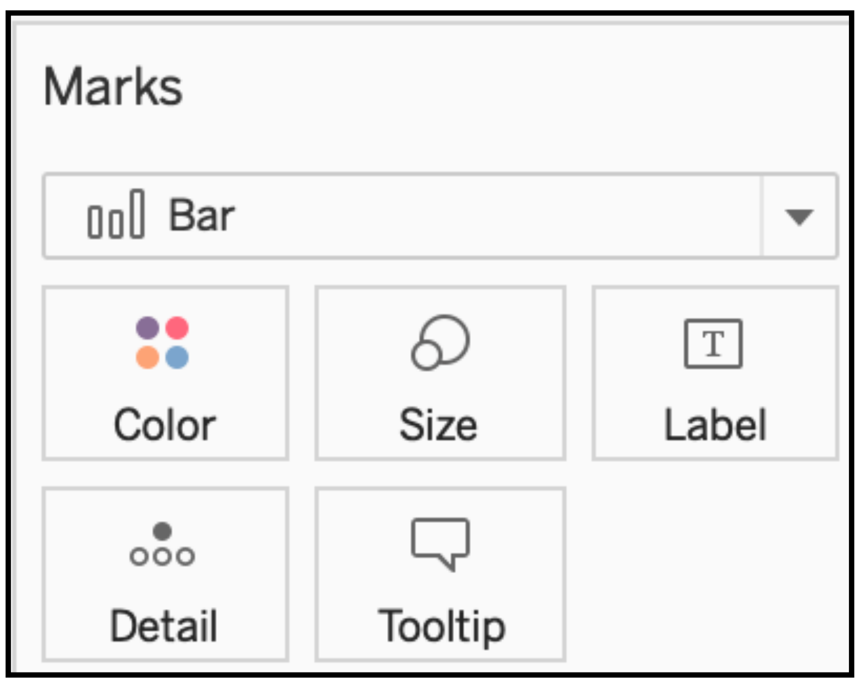
Tableau - Book1
File Data Worksheet Dashboard Story Analysis Map Format Server Window Help
Standard
Show Me
Data
Analytics
Orders+ (Multiple Con...
Dimensions
Customer ID
Customer Name
Market
Order Date
Order ID
Order Priority
Postal Code
Product ID
Product Name
Region
Row ID
Segment
Ship Date
Ship Mode
Measures
Discount
Profit
Quantity
Sales
Shipping Cost
Latitude (generated)
Longitude (generated)
Number of Records
Measure Values
Pages
Filters
Marks
Automatic
Color
Size
Label
Detail
Tooltip
Path
YEAR(Order..
Columns
MONTH(Order Date)
Rows
SUM(Sales)
Sheet 1
Order Date
Sales
500K
400K
300K
200K
100K
0K
January
February
March
April
May
June
July
August
September
October
November
December
YEAR(Order Date)
2012
2013
2014
2015
Data Source
Sheet 1
48 marks
1 row by 12 columns
SUM(Sales): 12,642,502
Lari

(Quick) Table Calculations

- **Query the data before building the graphic**

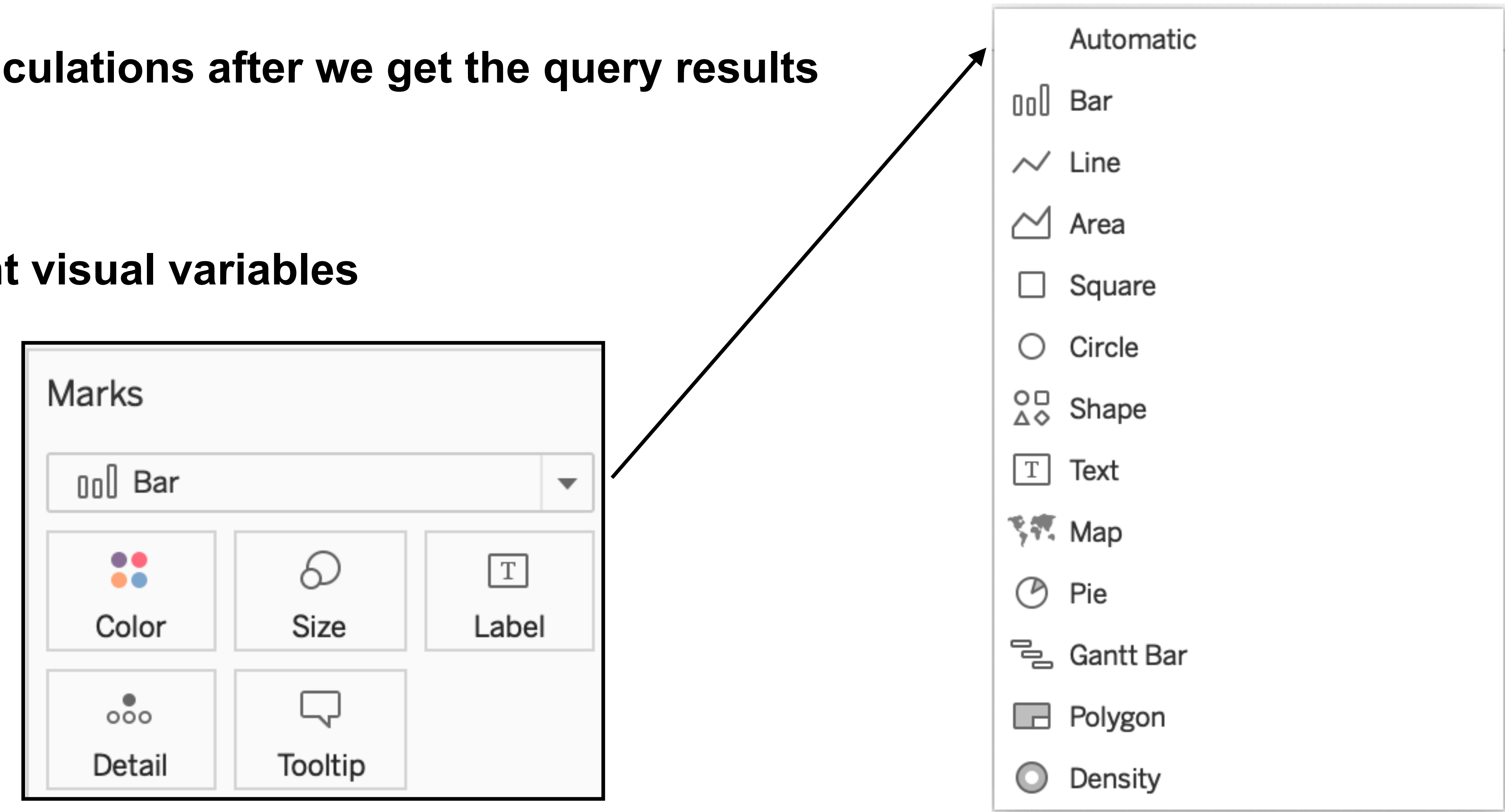
- **New calculations after we get the query results**

- **Different visual variables**



(Quick) Table Calculations

- **Query the data before building the graphic**
- **New calculations after we get the query results**
- **Different visual variables**



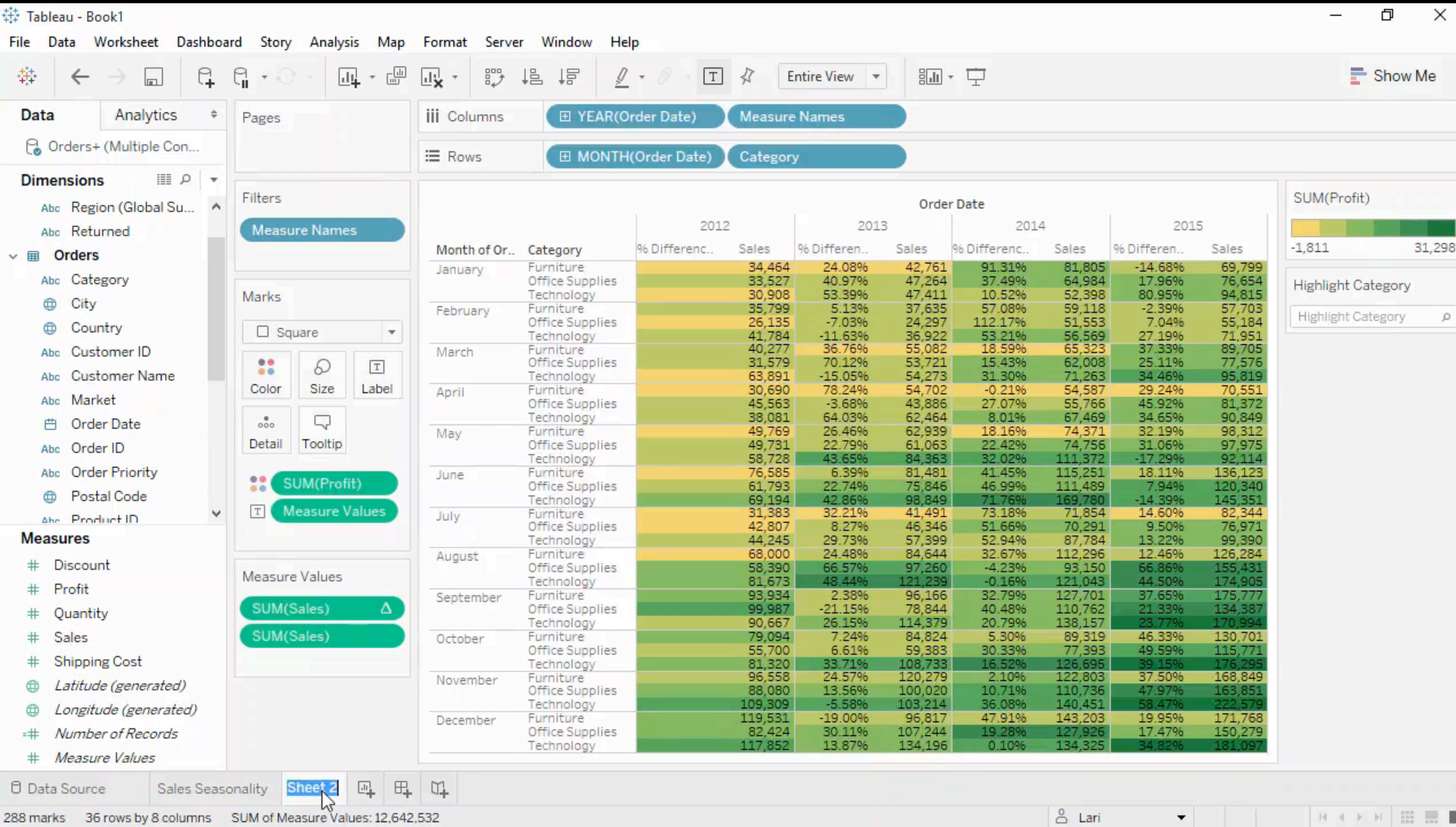


Tableau - Book1
File Data Worksheet Dashboard Story Analysis Map Format Server Window Help
Entire View
Show Me
Data Analytics
Orders+ (Multiple Con...
Dimensions
Region (Global Su...
Returned
Orders
Category
City
Country
Customer ID
Customer Name
Market
Order Date
Order ID
Order Priority
Postal Code
Product ID
Measures
Discount
Profit
Quantity
Sales
Shipping Cost
Latitude (generated)
Longitude (generated)
Number of Records
Measure Values
Pages
Filters
Measure Names
Marks
Square
Color Size Label
Detail Tooltip
SUM(Profit)
Measure Values
Measure Values
SUM(Sales)
SUM(Sales)
Columns YEAR(Order Date) Measure Names
Rows MONTH(Order Date) Category
Order Date
2012 2013 2014 2015
Month of Or.. Category % Differenc.. Sales % Differen.. Sales % Differenc.. Sales % Differen.. Sales
January Furniture 34,464 24.08% 42,761 91.31% 81,805 -14.68% 69,799
Office Supplies 33,527 40.97% 47,264 37.49% 64,984 17.96% 76,654
Technology 30,908 53.39% 47,411 10.52% 52,398 80.95% 94,815
February Furniture 35,799 5.13% 37,635 57.08% 59,118 -2.39% 57,703
Office Supplies 26,135 -7.03% 24,297 112.17% 51,553 7.04% 55,184
Technology 41,784 -11.63% 36,922 53.21% 56,569 27.19% 71,951
March Furniture 40,277 36.76% 55,082 18.59% 65,323 37.33% 89,705
Office Supplies 31,579 70.12% 53,721 15.43% 62,008 25.11% 77,576
Technology 63,891 -15.05% 54,273 31.30% 71,263 34.46% 95,819
April Furniture 30,690 78.24% 54,702 -0.21% 54,587 29.24% 70,551
Office Supplies 45,563 -3.68% 43,886 27.07% 55,766 45.92% 81,372
Technology 38,081 64.03% 62,464 8.01% 67,469 34.65% 90,849
May Furniture 49,769 26.46% 62,939 18.16% 74,371 32.19% 98,312
Office Supplies 49,731 22.79% 61,063 22.42% 74,756 31.06% 97,975
Technology 58,728 43.65% 84,363 32.02% 111,372 -17.29% 92,114
June Furniture 76,585 6.39% 81,481 41.45% 115,251 18.11% 136,123
Office Supplies 61,793 22.74% 75,846 46.99% 111,489 7.94% 120,340
Technology 69,194 42.86% 98,849 71.76% 169,780 -14.39% 145,351
July Furniture 31,383 32.21% 41,491 73.18% 71,854 14.60% 82,344
Office Supplies 42,807 8.27% 46,346 51.66% 70,291 9.50% 76,971
Technology 44,245 29.73% 57,399 52.94% 87,784 13.22% 99,390
August Furniture 68,000 24.48% 84,644 32.67% 112,296 12.46% 126,284
Office Supplies 58,390 66.57% 97,260 -4.23% 93,150 66.86% 155,431
Technology 81,673 48.44% 121,239 -0.16% 121,043 44.50% 174,905
September Furniture 93,934 2.38% 96,166 32.79% 127,701 37.65% 175,777
Office Supplies 99,987 -21.15% 78,844 40.48% 110,762 21.33% 134,387
Technology 90,667 26.15% 114,379 20.79% 138,157 23.77% 170,994
October Furniture 79,094 7.24% 84,824 5.30% 89,319 46.33% 130,701
Office Supplies 55,700 6.61% 59,383 30.33% 77,393 49.59% 115,771
Technology 81,320 33.71% 108,733 16.52% 126,695 39.15% 176,295
November Furniture 96,558 24.57% 120,279 2.10% 122,803 37.50% 168,849
Office Supplies 88,080 13.56% 100,020 10.71% 110,736 47.97% 163,851
Technology 109,309 -5.58% 103,214 36.08% 140,451 58.47% 222,579
December Furniture 119,531 -19.00% 96,817 47.91% 143,203 19.95% 171,768
Office Supplies 82,424 30.11% 107,244 19.28% 127,926 17.47% 150,279
Technology 117,852 13.87% 134,196 0.10% 134,325 34.82% 181,097
SUM(Profit)
-1,811 31,298
Highlight Category
Highlight Category
Data Source Sales Seasonality Sheet 2
288 marks 36 rows by 8 columns SUM of Measure Values: 12,642,532
Lari

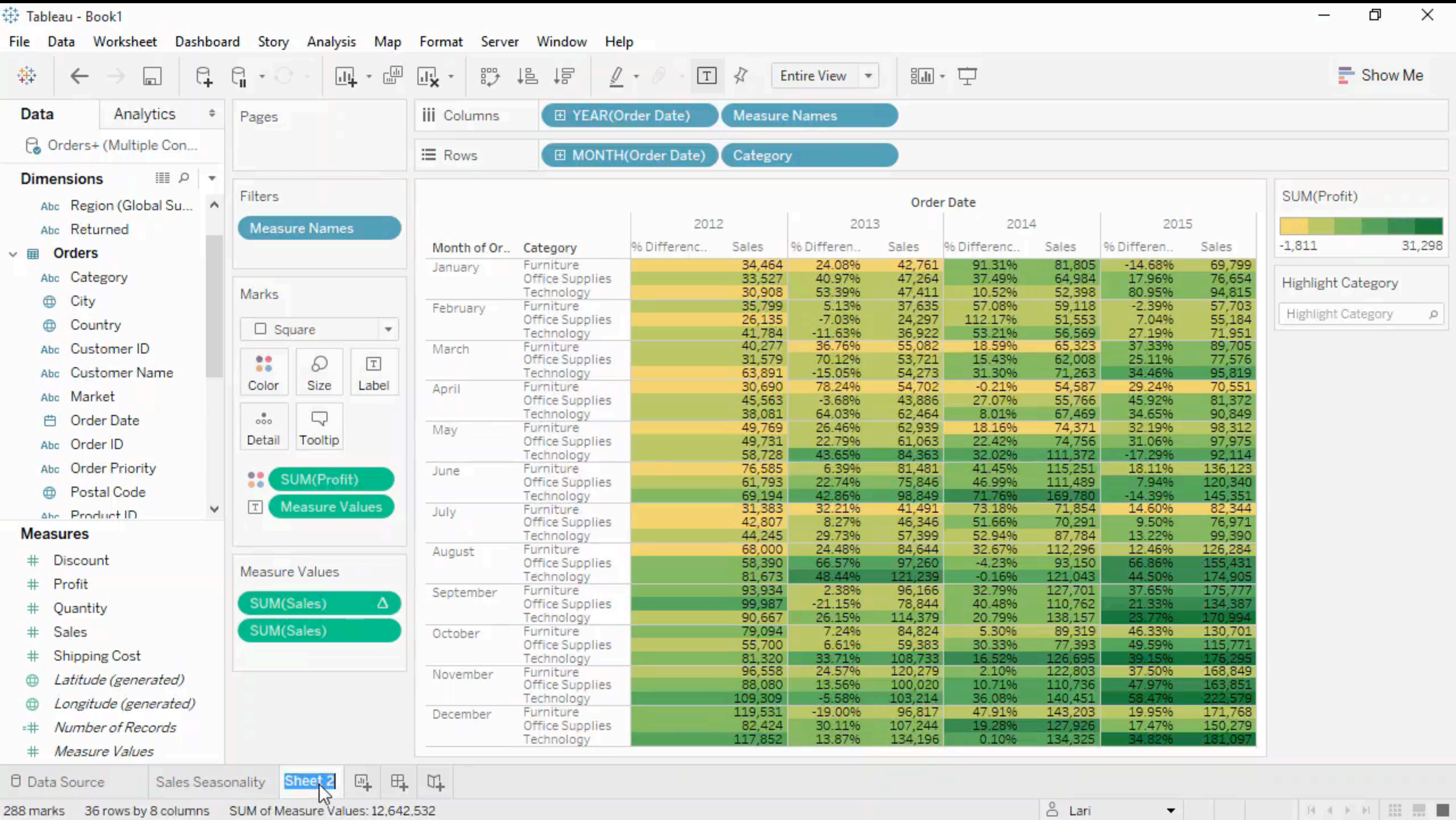


Tableau - Book1
File Data Worksheet Dashboard Story Analysis Map Format Server Window Help
Entire View
Show Me
Data Analytics
Orders+ (Multiple Con...
Dimensions
Region (Global Su...
Returned
Orders
Category
City
Country
Customer ID
Customer Name
Market
Order Date
Order ID
Order Priority
Postal Code
Product ID
Measures
Discount
Profit
Quantity
Sales
Shipping Cost
Latitude (generated)
Longitude (generated)
Number of Records
Measure Values
Pages
Filters
Measure Names
Marks
Square
Color Size Label
Detail Tooltip
SUM(Profit)
Measure Values
Measure Values
SUM(Sales) Δ
SUM(Sales)
Columns YEAR(Order Date) Measure Names
Rows MONTH(Order Date) Category
Order Date
2012 2013 2014 2015
Month of Or.. Category % Differenc.. Sales % Differen.. Sales % Differenc.. Sales % Differen.. Sales
January Furniture 34,464 24.08% 42,761 91.31% 81,805 -14.68% 69,799
Office Supplies 33,527 40.97% 47,264 37.49% 64,984 17.96% 76,654
Technology 30,908 53.39% 47,411 10.52% 52,398 80.95% 94,815
February Furniture 35,799 5.13% 37,635 57.08% 59,118 -2.39% 57,703
Office Supplies 26,135 -7.03% 24,297 112.17% 51,553 7.04% 55,184
Technology 41,784 -11.63% 36,922 53.21% 56,569 27.19% 71,951
March Furniture 40,277 36.76% 55,082 18.59% 65,323 37.33% 89,705
Office Supplies 31,579 70.12% 53,721 15.43% 62,008 25.11% 77,576
Technology 63,891 -15.05% 54,273 31.30% 71,263 34.46% 95,819
April Furniture 30,690 78.24% 54,702 -0.21% 54,587 29.24% 70,551
Office Supplies 45,563 -3.68% 43,886 27.07% 55,766 45.92% 81,372
Technology 38,081 64.03% 62,464 8.01% 67,469 34.65% 90,849
May Furniture 49,769 26.46% 62,939 18.16% 74,371 32.19% 98,312
Office Supplies 49,731 22.79% 61,063 22.42% 74,756 31.06% 97,975
Technology 58,728 43.65% 84,363 32.02% 111,372 -17.29% 92,114
June Furniture 76,585 6.39% 81,481 41.45% 115,251 18.11% 136,123
Office Supplies 61,793 22.74% 75,846 46.99% 111,489 7.94% 120,340
Technology 69,194 42.86% 98,849 71.76% 169,780 -14.39% 145,351
July Furniture 31,383 32.21% 41,491 73.18% 71,854 14.60% 82,344
Office Supplies 42,807 8.27% 46,346 51.66% 70,291 9.50% 76,971
Technology 44,245 29.73% 57,399 52.94% 87,784 13.22% 99,390
August Furniture 68,000 24.48% 84,644 32.67% 112,296 12.46% 126,284
Office Supplies 58,390 66.57% 97,260 -4.23% 93,150 66.86% 155,431
Technology 81,673 48.44% 121,239 -0.16% 121,043 44.50% 174,905
September Furniture 93,934 2.38% 96,166 32.79% 127,701 37.65% 175,777
Office Supplies 99,987 -21.15% 78,844 40.48% 110,762 21.33% 134,387
Technology 90,667 26.15% 114,379 20.79% 138,157 23.77% 170,994
October Furniture 79,094 7.24% 84,824 5.30% 89,319 46.33% 130,701
Office Supplies 55,700 6.61% 59,383 30.33% 77,393 49.59% 115,771
Technology 81,320 33.71% 108,733 16.52% 126,695 39.15% 176,295
November Furniture 96,558 24.57% 120,279 2.10% 122,803 37.50% 168,849
Office Supplies 88,080 13.56% 100,020 10.71% 110,736 47.97% 163,851
Technology 109,309 -5.58% 103,214 36.08% 140,451 58.47% 222,579
December Furniture 119,531 -19.00% 96,817 47.91% 143,203 19.95% 171,768
Office Supplies 82,424 30.11% 107,244 19.28% 127,926 17.47% 150,279
Technology 117,852 13.87% 134,196 0.10% 134,325 34.82% 181,097
SUM(Profit)
-1,811 31,298
Highlight Category
Highlight Category
Data Source Sales Seasonality Sheet 2
288 marks 36 rows by 8 columns SUM of Measure Values: 12,642,532
Lari

Map visualizations - ShowMe

- **Select some Dimensions and Measures and see possible graphics**

- **Maps based on names of countries and cities !**

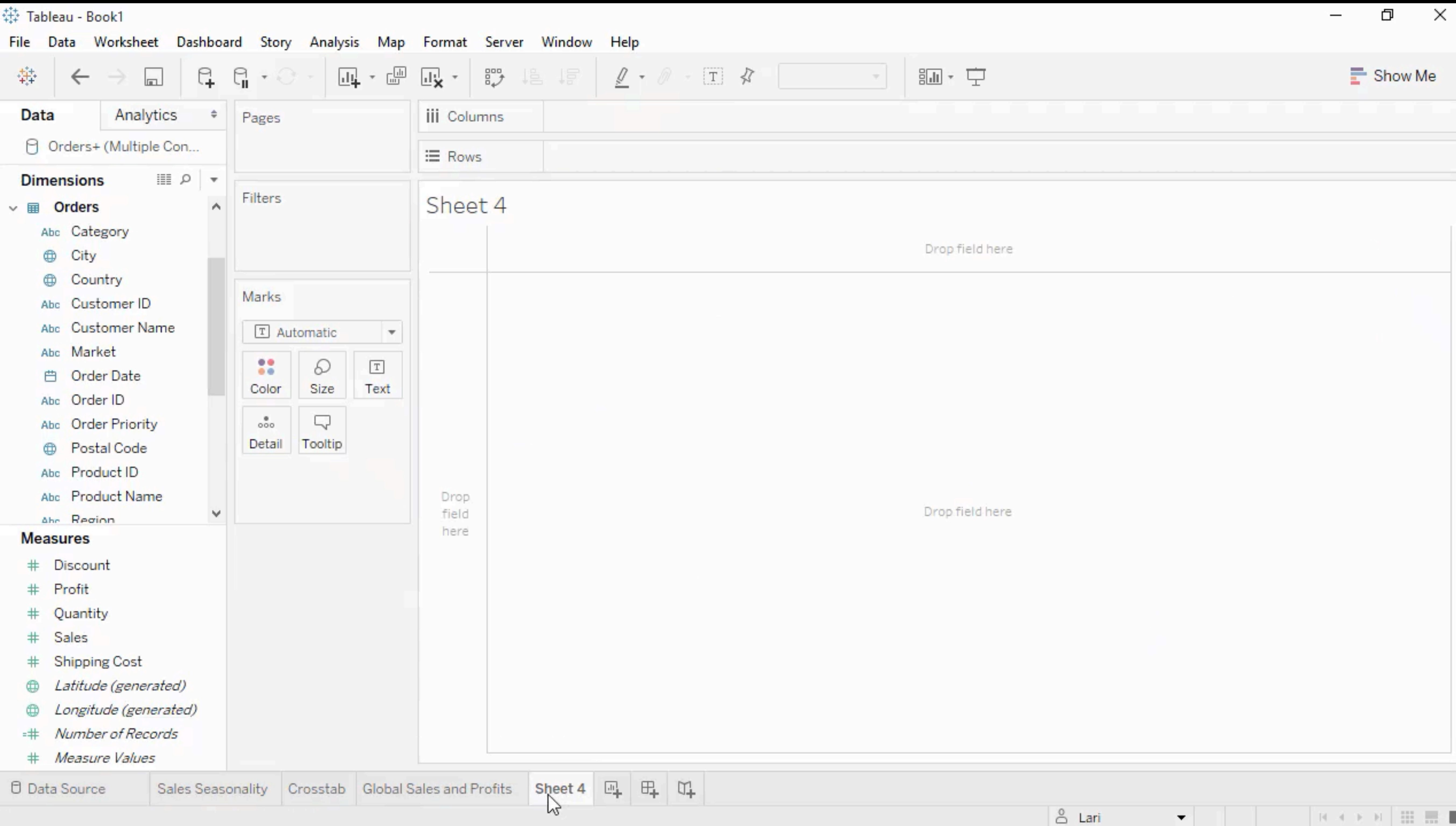


Tableau - Book1
File
Data
Worksheet
Dashboard
Story
Analysis
Map
Format
Server
Window
Help
Show Me
Data
Analytics
Orders+ (Multiple Con...
Dimensions
Orders
Category
City
Country
Customer ID
Customer Name
Market
Order Date
Order ID
Order Priority
Postal Code
Product ID
Product Name
Region
Measures
Discount
Profit
Quantity
Sales
Shipping Cost
Latitude (generated)
Longitude (generated)
Number of Records
Measure Values
Pages
Filters
Marks
Automatic
Color
Size
Text
Detail
Tooltip
Columns
Rows
Sheet 4
Drop field here
Drop field here
Drop field here
Data Source
Sales Seasonality
Crosstab
Global Sales and Profits
Sheet 4
Lari

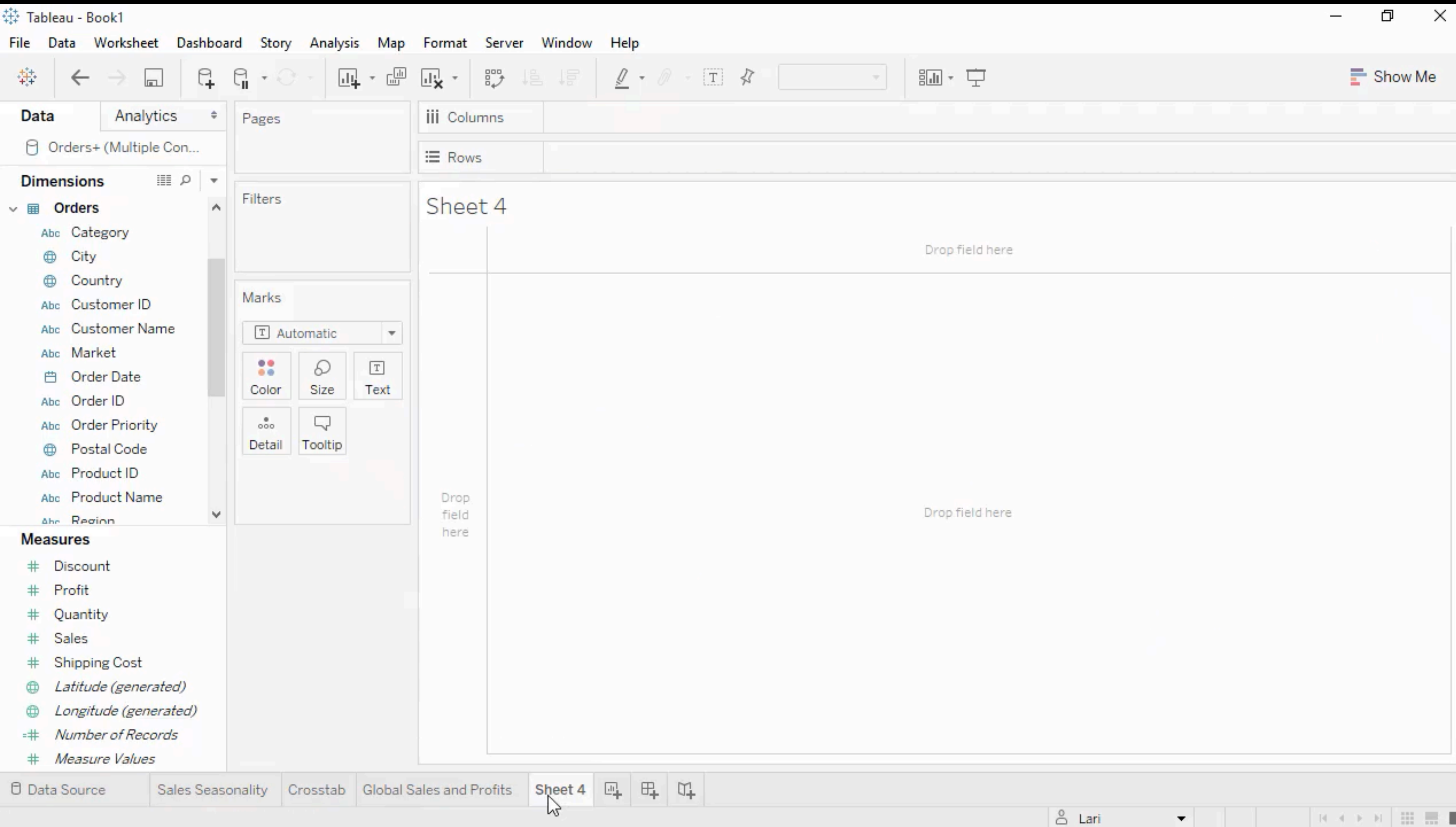


Tableau - Book1
File
Data
Worksheet
Dashboard
Story
Analysis
Map
Format
Server
Window
Help
Show Me
Data
Analytics
Orders+ (Multiple Con...
Dimensions
Orders
Category
City
Country
Customer ID
Customer Name
Market
Order Date
Order ID
Order Priority
Postal Code
Product ID
Product Name
Region
Measures
Discount
Profit
Quantity
Sales
Shipping Cost
Latitude (generated)
Longitude (generated)
Number of Records
Measure Values
Pages
Filters
Marks
Automatic
Color
Size
Text
Detail
Tooltip
Columns
Rows
Sheet 4
Drop field here
Drop field here
Drop field here
Data Source
Sales Seasonality
Crosstab
Global Sales and Profits
Sheet 4
Lari

More ShowMe and Hierarchies

- **Hierarchies as sequences of dimensions**
  - **Drill Down and Drill up**
- **Sort !**
- **Grouping Data**

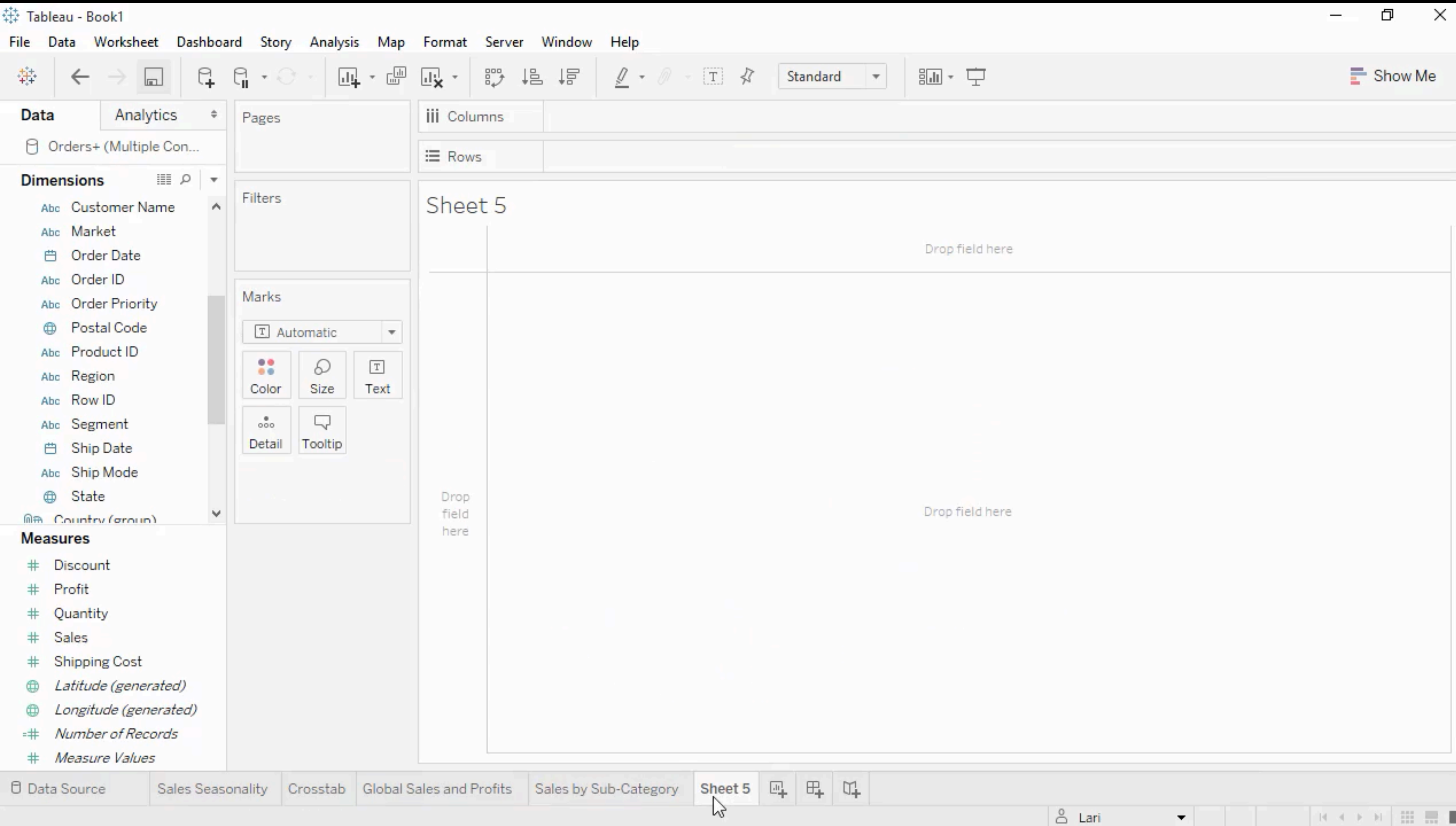


Tableau - Book1
File
Data
Worksheet
Dashboard
Story
Analysis
Map
Format
Server
Window
Help
Standard
Show Me
Data
Analytics
Orders+ (Multiple Con...
Pages
Columns
Rows
Filters
Marks
Automatic
Color
Size
Text
Detail
Tooltip
Sheet 5
Drop field here
Drop field here
Drop field here
Dimensions
Customer Name
Market
Order Date
Order ID
Order Priority
Postal Code
Product ID
Region
Row ID
Segment
Ship Date
Ship Mode
State
Country (group)
Measures
Discount
Profit
Quantity
Sales
Shipping Cost
Latitude (generated)
Longitude (generated)
Number of Records
Measure Values
Data Source
Sales Seasonality
Crosstab
Global Sales and Profits
Sales by Sub-Category
Sheet 5
Lari

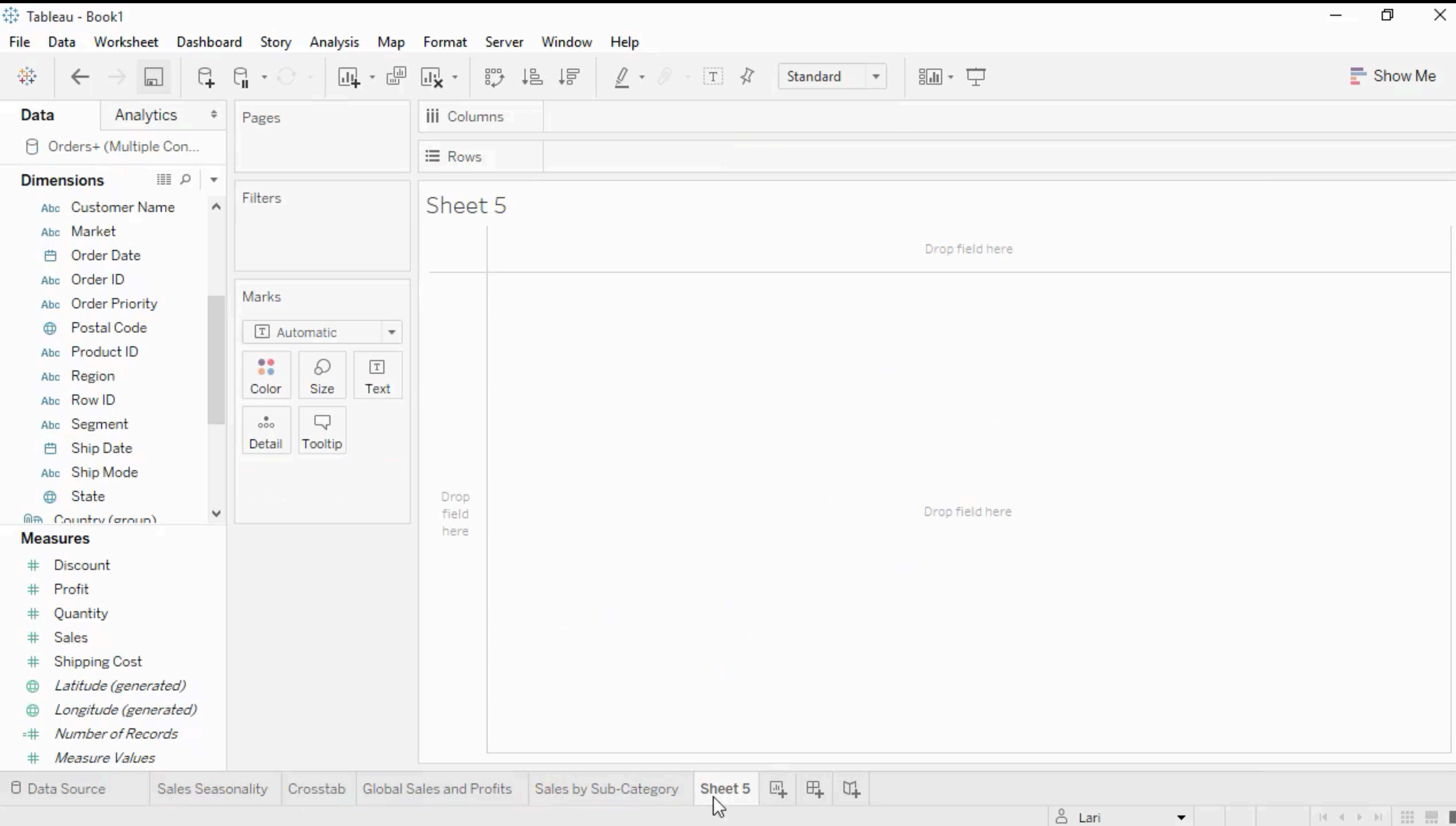


Tableau - Book1
File
Data
Worksheet
Dashboard
Story
Analysis
Map
Format
Server
Window
Help
Standard
Show Me
Data
Analytics
Orders+ (Multiple Con...
Pages
Columns
Rows
Filters
Dimensions
Customer Name
Market
Order Date
Order ID
Order Priority
Postal Code
Product ID
Region
Row ID
Segment
Ship Date
Ship Mode
State
Country (group)
Marks
Automatic
Color
Size
Text
Detail
Tooltip
Sheet 5
Drop field here
Drop field here
Drop field here
Measures
Discount
Profit
Quantity
Sales
Shipping Cost
Latitude (generated)
Longitude (generated)
Number of Records
Measure Values
Data Source
Sales Seasonality
Crosstab
Global Sales and Profits
Sales by Sub-Category
Sheet 5
Lari

Scatter graphics

- **Level of detail**
- **Using different visual variables**
- **Analytics**
  - **Trend lines**
- **See the underline data**

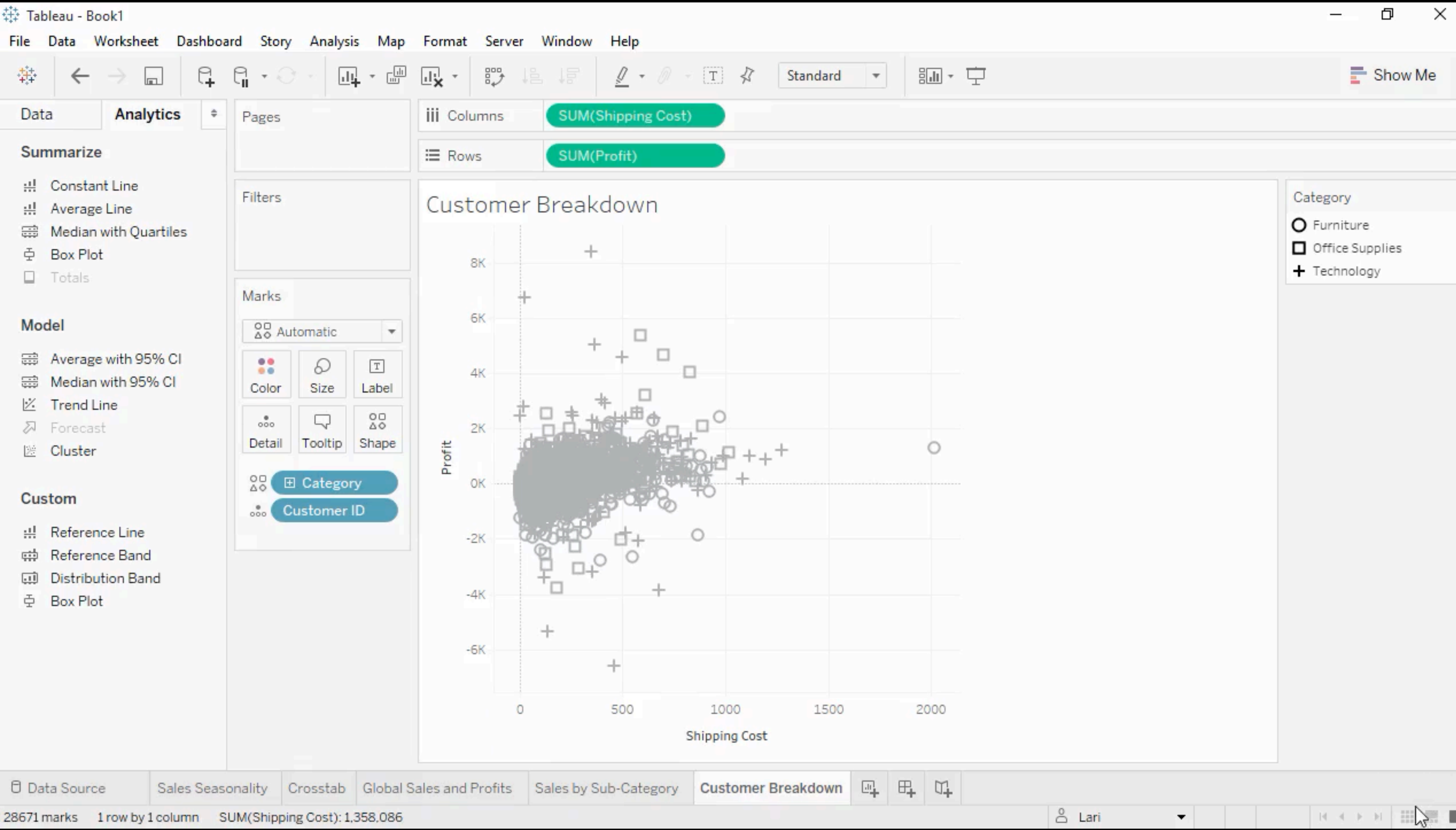


Tableau - Book1
File Data Worksheet Dashboard Story Analysis Map Format Server Window Help
Standard
Show Me
Data
Analytics
Summarize
Constant Line
Average Line
Median with Quartiles
Box Plot
Totals
Model
Average with 95% CI
Median with 95% CI
Trend Line
Forecast
Cluster
Custom
Reference Line
Reference Band
Distribution Band
Box Plot
Pages
Filters
Marks
Automatic
Color
Size
Label
Detail
Tooltip
Shape
Category
Customer ID
Columns
SUM(Shipping Cost)
Rows
SUM(Profit)
Customer Breakdown
Profit
Shipping Cost
Category
Furniture
Office Supplies
Technology
Data Source
Sales Seasonality
Crosstab
Global Sales and Profits
Sales by Sub-Category
Customer Breakdown
28671 marks
1 row by 1 column
SUM(Shipping Cost): 1,358,086
Lari

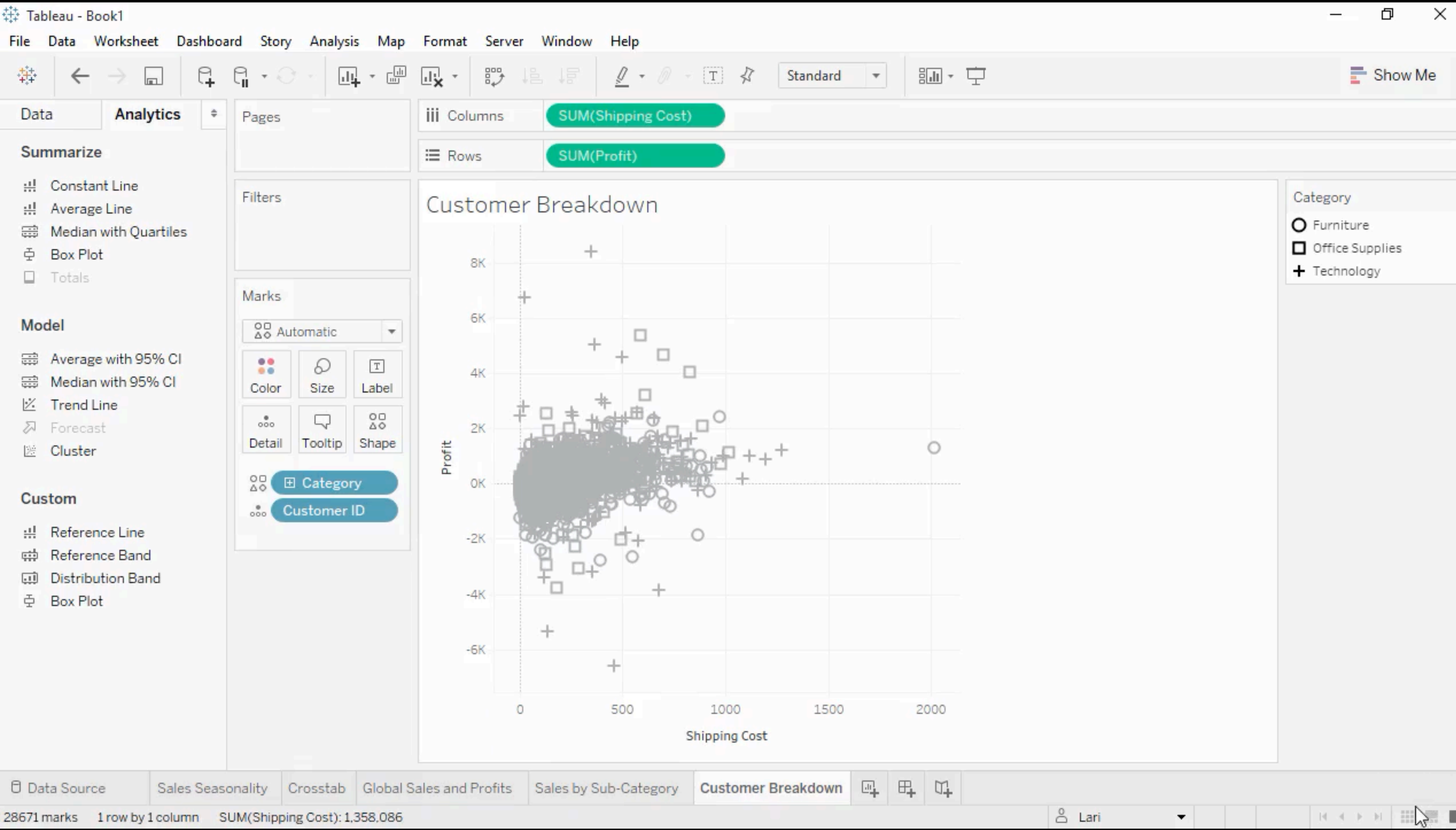


Tableau - Book1
File Data Worksheet Dashboard Story Analysis Map Format Server Window Help
Standard
Show Me
Data Analytics
Summarize
Constant Line
Average Line
Median with Quartiles
Box Plot
Totals
Model
Average with 95% CI
Median with 95% CI
Trend Line
Forecast
Cluster
Custom
Reference Line
Reference Band
Distribution Band
Box Plot
Pages
Filters
Marks
Automatic
Color Size Label
Detail Tooltip Shape
Category
Customer ID
Columns SUM(Shipping Cost)
Rows SUM(Profit)
Customer Breakdown
Profit
8K 6K 4K 2K 0K -2K -4K -6K
0 500 1000 1500 2000
Shipping Cost
Category
Furniture
Office Supplies
Technology
Data Source Sales Seasonality Crosstab Global Sales and Profits Sales by Sub-Category Customer Breakdown
28671 marks 1 row by 1 column SUM(Shipping Cost): 1,358,086
Lari

Dashboards and Stories

- **Dashboard**
  - **Combining multiple views**
  - **Linking the views**
  - **Filtering based on one view**

Dashboards and Stories

- **Dashboard**
  - **Combining multiple views**
  - **Linking the views**
  - **Filtering based on one view**

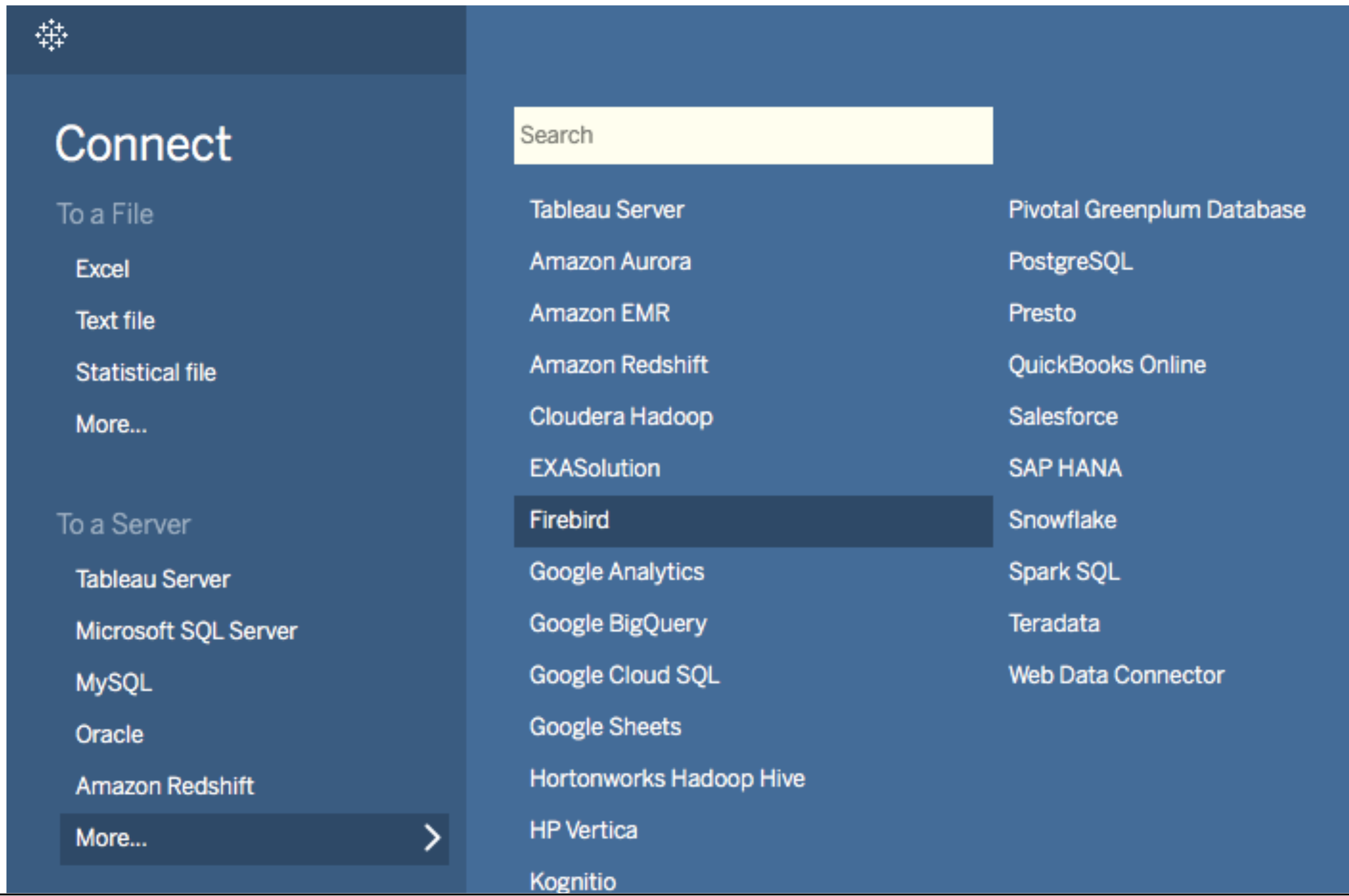
- **Story**
  - **Steps**
  - **Highlights**
  - **Tell a story**

Main Concepts

Data: Data Sources and Data Types

Data Sources

- Tableau allows you to connect to several data sources in different formats
  - Files: Excel, text, Statistical, …
  - Servers…



Connecting to Data

- **The first step is to connect to one or more data sources**
  - **It is possible to join data from different sources and perform joins based on common fields:**

Table 1

ID	First Name	Last Name	Publisher Type
20034	Adam	Davis	Independent
20165	Ashley	Garcia	Big
20233	Susan	Nguyen	Small/medium

+

Table 2

Book Title	Price	Royalty	ID
Weather in the Alps	19.99	5,000	20165
My Physics	8.99	3,500	20800
The Magic Shoe Lace	15.99	7,000	20034

ID	First Name	Last Name	Publisher Type	Book Title	Price	Royalty
20034	Adam	Davis	Independent	The Magic Shoe Lace	15.99	7,000
20165	Ashley	Garcia	Big	Weather in the Alps	19.99	5,000

Connecting to Data

- **All kinds of joins are possible…**

Join Type	Result	Description
Inner	When you use an inner join to combine tables, the result is a table that contains values that have matches in both tables.	
Left	When you use a left join to combine tables, the result is a table that contains all values from the left table and corresponding matches from the right table. When a value in the left table doesn't have a corresponding match in the right table, you see a null value in the data grid.	
Right	When you use a right join to combine tables, the result is a table that contains all values from the right table and corresponding matches from the left table. When a value in the right table doesn't have a corresponding match in the left table, you see a null value in the data grid.	
Full outer	When you use a full outer join to combine tables, the result is a table that contains all values from both tables. When a value from either table doesn't have a match with the other table, you see a null value in the data grid.	
Union	Though union is not a type of join, union is another method for combining two or more tables by appending rows of data from one table to another. Ideally, the tables that you union have the same number of fields, and those fields have matching names and data types. For more information about union, see Union Your Data.	

see more at: https://onlinehelp.tableau.com/current/pro/desktop/en-us/joining_tables.html

Connecting to Data

- **A simple example: Cars dataset**

```
NAME:  2004 New Car and Truck Data
TYPE:  Sample
SIZE:  428 observations, 19 variables

VARIABLE DESCRIPTIONS:

Columns   Variables
  1- 45   Vehicle Name
 47                Sports Car? (1=yes, 0=no)
 49                Sport Utility Vehicle? (1=yes, 0=no)
 51                Wagon? (1=yes, 0=no)
 53                Minivan? (1=yes, 0=no)
 55                Pickup? (1=yes, 0=no)
 57                All-Wheel Drive? (1=yes, 0=no)
 59                Rear-Wheel Drive? (1=yes, 0=no)
 61- 66   Suggested Retail Price, what the manufacturer thinks the
                   vehicle is worth, including adequate profit for the
                   automaker and the dealer (U.S. Dollars)
 68- 73   Dealer Cost (or "invoice price"), what the dealership pays
                   the manufacturer (U.S. Dollars)
 75- 77   Engine Size (liters)
 79- 80   Number of Cylinders (=-1 if rotary engine)
 82- 84   Horsepower
 86- 87   City Miles Per Gallon
 89- 90   Highway Miles Per Gallon
 92- 95   Weight (Pounds)
 97- 99   Wheel Base (inches)
101-103   Length (inches)
105-106   Width (inches)
```

04cars data

A248 — Mazda 3 i 4dr

	Vehicle Name	Small/Sporty/ Compact/Large Sedan	Sports Car	SUV	Wagon	Minivan	Pickup	AWD	RWD	Retail Price	Dealer Cost	Engine Size (l)	Cyl	HP	City MPG	Hwy MPG	Weight	Wheel Base	Len	Width
250	Mazda 6 i 4dr	1	0	0	0	0	0	0	0	19270	17817	2,3	4	160	24	32	3042	105	187	70
251	Mercedes-Benz C230 Sport 2dr	1	0	0	0	0	0	0	1	26060	24249	1,8	4	189	22	30	3250	107	178	68
252	Mercedes-Benz C240	0	0	0	1	0	0	0	1	33780	31466	2,6	6	168	19	25	3470	107	179	68
253	Mercedes-Benz C240 4dr	1	0	0	0	0	0	0	1	32280	30071	2,6	6	168	20	25	3360	107	178	68
254	Mercedes-Benz C240 4dr	1	0	0	0	0	0	1	0	33480	31187	2,6	6	168	19	25	3360	107	178	68
255	Mercedes-Benz C32 AMG 4dr	1	0	0	0	0	0	0	1	52120	48522	3,2	6	349	16	21	3540	107	178	68
256	Mercedes-Benz C320 4dr	1	0	0	0	0	0	0	1	37630	35046	3,2	6	215	20	26	3450	107	178	68
257	Mercedes-Benz C320 4dr	1	0	0	0	0	0	1	0	38830	36162	3,2	6	215	19	27	*	107	178	68
258	Mercedes-Benz C320 Sport 2dr	1	0	0	0	0	0	0	1	28370	26435	3,2	6	215	19	26	3430	107	178	68
259	Mercedes-Benz C320 Sport 4dr	1	0	0	0	0	0	0	1	35920	33456	3,2	6	215	19	26	3430	107	178	68
260	Mercedes-Benz CL500 2dr	1	0	0	0	0	0	0	1	94820	88324	5	8	302	16	24	4085	114	196	73
261	Mercedes-Benz CL600 2dr	1	0	0	0	0	0	0	1	1E+05	119600	5,5	12	493	13	19	4473	114	196	73
262	Mercedes-Benz CLK320 coupe 2dr (convertible)	1	0	0	0	0	0	0	1	45707	41966	3,2	6	215	20	26	3770	107	183	69
263	Mercedes-Benz CLK500 coupe 2dr (convertible)	1	0	0	0	0	0	0	1	52800	49104	5	8	302	17	22	3585	107	183	69
264	Mercedes-Benz E320	0	0	0	1	0	0	0	1	50670	47174	3,2	6	221	19	27	3966	112	190	71
265	Mercedes-Benz E320 4dr	1	0	0	0	0	0	0	1	48170	44849	3,2	6	221	19	27	3635	112	190	71
266	Mercedes-Benz E500	0	0	0	1	0	0	1	0	60670	56474	5	8	302	16	24	4230	112	190	71
267	Mercedes-Benz E500 4dr	1	0	0	0	0	0	0	1	57270	53382	5	8	302	16	20	3815	112	190	71
268	Mercedes-Benz G500	0	0	1	0	0	0	1	0	76870	71540	5	8	292	13	14	5423	112	186	71
269	Mercedes-Benz ML500	0	0	1	0	0	0	1	0	46470	43268	5	8	288	14	17	4874	111	183	72
270	Mercedes-Benz S430 4dr	1	0	0	0	0	0	0	1	74320	69168	4,3	8	275	18	26	4160	122	203	73
271	Mercedes-Benz S500 4dr	1	0	0	0	0	0	1	0	86970	80939	5	8	302	16	24	4390	122	203	73
272	Mercedes-Benz SL500 convertible 2dr	0	1	0	0	0	0	0	1	90520	84325	5	8	302	16	23	4065	101	179	72
273	Mercedes-Benz SL55 AMG 2dr	0	1	0	0	0	0	0	1	1E+05	113388	5,5	8	493	14	21	4235	101	179	72
274	Mercedes-Benz SL600 convertible 2dr	0	1	0	0	0	0	0	1	1E+05	117854	5,5	12	493	13	19	4429	101	179	72
275	Mercedes-Benz SLK230 convertible 2dr	0	1	0	0	0	0	0	1	40320	37548	2,3	4	192	21	29	3055	95	158	68
276	Mercedes-Benz SLK32 AMG 2dr	0	1	0	0	0	0	0	1	56170	52289	3,2	6	349	17	22	3220	95	158	68
277	Mercury Grand Marquis GS 4dr	1	0	0	0	0	0	0	1	24695	23217	4,6	8	224	17	25	4052	115	212	78
278	Mercury Grand Marquis LS Premium 4dr	1	0	0	0	0	0	0	1	29595	27148	4,6	8	224	17	25	4052	115	212	78
279	Mercury Grand Marquis LS Ultimate 4dr	1	0	0	0	0	0	0	1	30895	28318	4,6	8	224	17	25	4052	115	212	78
280	Mercury Marauder 4dr	1	0	0	0	0	0	0	1	34495	31558	4,6	8	302	17	23	4195	115	212	78

04cars

Tableau Data Source View

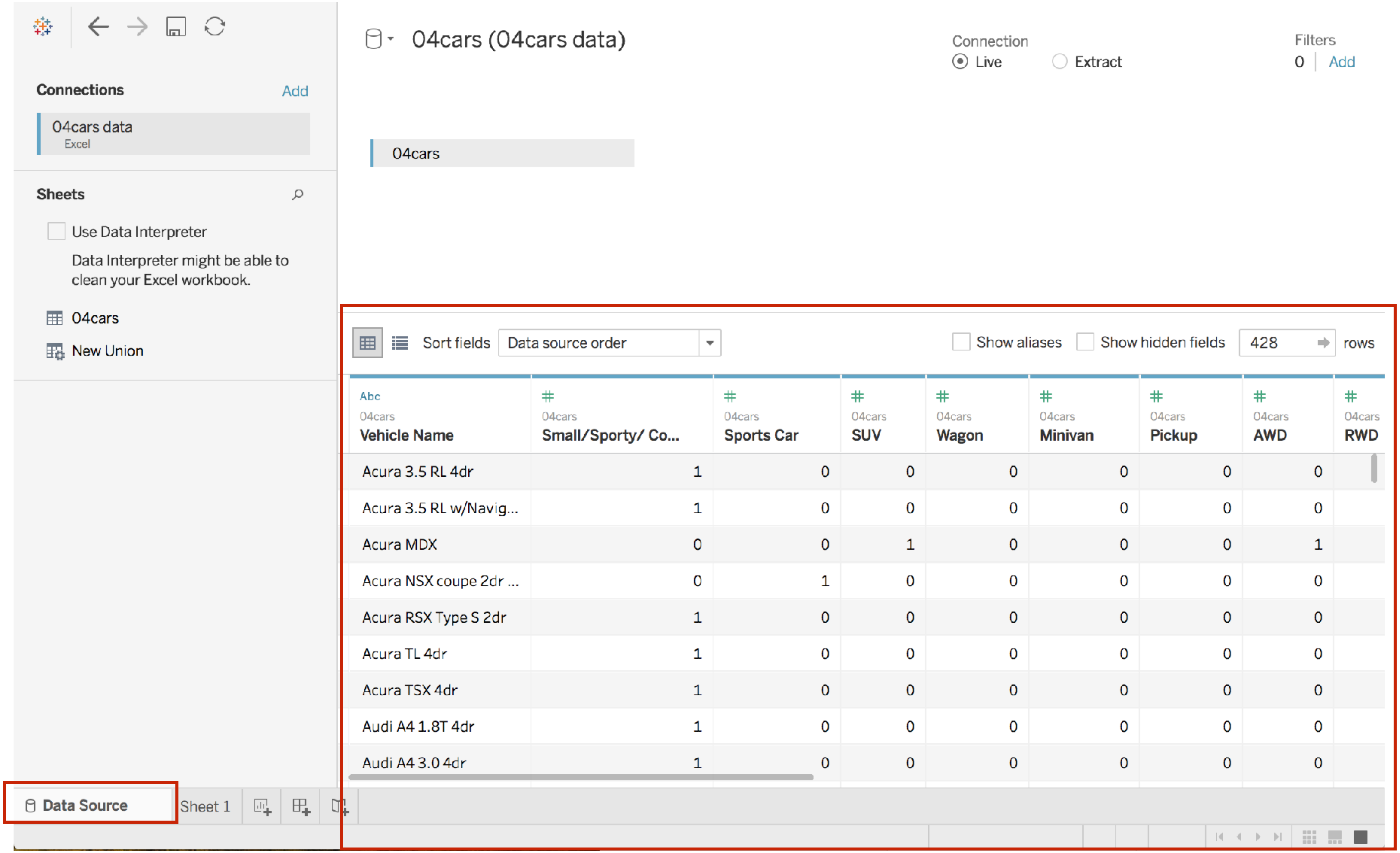


Tableau Data Source View

- **A simple example: Cars dataset**

Variable name

data type

Abc 04cars **Vehicle Name**	# 04cars **Small/Sporty/ Co...**	# 04cars **Sports Car**	# 04cars **SUV**	# 04cars **Wagon**	# 04cars **Minivan**	# 04cars **Pickup**	# 04cars **AWD**	# 04cars **RWD**	# 04cars **Retail Price**
Acura 3.5 RL 4dr	1	0	0	0	0	0	0	0	43,755
Acura 3.5 RL w/Navig...	1	0	0	0	0	0	0	0	46,100
Acura MDX	0	0	1	0	0	0	1	0	36,945
Acura NSX coupe 2dr ...	0	1	0	0	0	0	0	1	89,765
Acura RSX Type S 2dr	1	0	0	0	0	0	0	0	23,820
Acura TL 4dr	1	0	0	0	0	0	0	0	33,195
Acura TSX 4dr	1	0	0	0	0	0	0	0	26,990
Audi A4 1.8T 4dr	1	0	0	0	0	0	0	0	25,940
Audi A4 3.0 4dr	1	0	0	0	0	0	0	0	31,840
Audi A4 3.0 converti...	1	0	0	0	0	0	0	0	42,490
Audi A4 3.0 Quattro ...	1	0	0	0	0	0	1	0	34,480
Audi A4 3.0 Quattro ...	1	0	0	0	0	0	1	0	33,430

Data Types

- **All fields in a data source have a data type. Tableau automatically infers datatypes from the values if the data source doesn't provide the necessary metadata**
- **Available data types:**
  - **Text/Strings**
  - **Dates**
  - **Date and Time**
  - **Numeric**
  - **Boolean**
  - **Geographic**

data types can be changed later

Icon	Data type
Abc	Text (string) values
	Date values
	Date & Time values
#	Numerical values
T|F	Boolean values (relational only)
	Geographic values (used with maps)

Data Types

- **Text/Strings - For categorical (ordinal or not) or nominal data**
  - “Tall/Short/Medium”, “yellow/red/green/blue”, “expensive/economic”, “North/South/West”, “Fernando/João/Maria/…”, “Food/Cleaning/Fruit/Cloth/Office/…”

Data Types

- **Text/Strings - For categorical (ordinal or not) or nominal data**
  - "Tall/Short/Medium", "yellow/red/green/blue", "expensive/economic", "North/South/West", "Fernando/João/Maria/…", "Food/Cleaning/Fruit/Cloth/Office/…"
- **Dates - Used to tag events in time with day/week/month/quarter/year granularities**

Data Types

- **Text/Strings - For categorical (ordinal or not) or nominal data**
  - “Tall/Short/Medium”, “yellow/red/green/blue”, “expensive/economic”, “North/South/West”, “Fernando/João/Maria/…”, “Food/Cleaning/Fruit/Cloth/Office/…”
- **Dates - Used to tag events in time with day/week/month/quarter/year granularities**
- **Date and Time - Date + Time (Hours, Minutes, Seconds, …)**

Data Types

- **Text/Strings - For categorical (ordinal or not) or nominal data**
  - “Tall/Short/Medium”, “yellow/red/green/blue”, “expensive/economic”, “North/South/West”, “Fernando/João/Maria/…”, “Food/Cleaning/Fruit/Cloth/Office/…”
- **Dates - Used to tag events in time with day/week/month/quarter/year granularities**
- **Date and Time - Date + Time (Hours, Minutes, Seconds, …)**
- **Numeric - numerical data**
  - **Counts, quantities, distances, $$$, anything that can be measured…**

Data Types

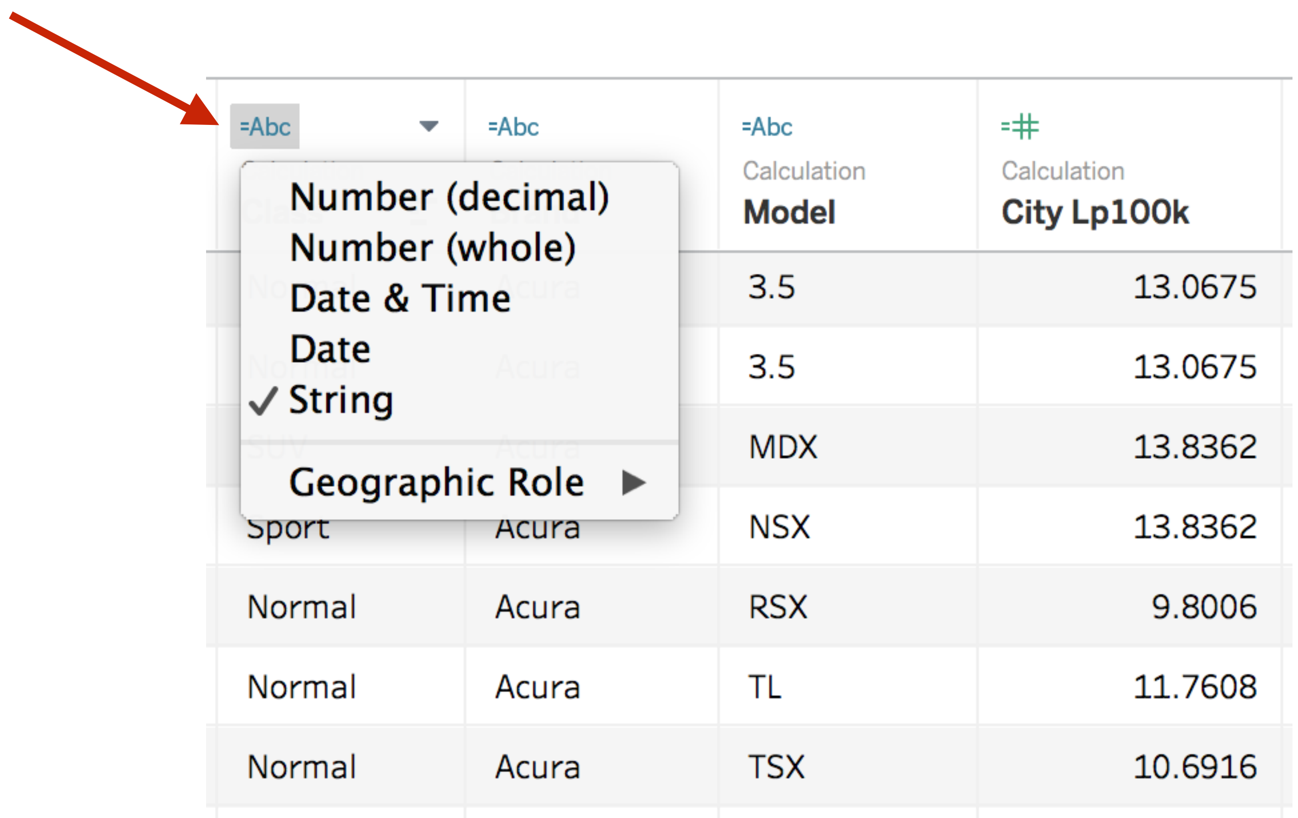
- **Text/Strings - For categorical (ordinal or not) or nominal data**
  - “Tall/Short/Medium”, “yellow/red/green/blue”, “expensive/economic”, “North/South/West”, “Fernando/João/Maria/…”, “Food/Cleaning/Fruit/Cloth/Office/…”
- **Dates - Used to tag events in time with day/week/month/quarter/year granularities**
- **Date and Time - Date + Time (Hours, Minutes, Seconds, …)**
- **Numeric - numerical data**
  - **Counts, quantities, distances, $$$, anything that can be measured…**
- **Boolean - Used to classify as True/False, Yes/No with respect to some property**
  - Garage? Breakfast included? Tall? Short? Expensive? 5-door?

Data Types

- **Text/Strings - For categorical (ordinal or not) or nominal data**
  - "Tall/Short/Medium", "yellow/red/green/blue", "expensive/economic", "North/South/West", "Fernando/João/Maria/…", "Food/Cleaning/Fruit/Cloth/Office/…"
- **Dates - Used to tag events in time with day/week/month/quarter/year granularities**
- **Date and Time - Date + Time (Hours, Minutes, Seconds, …)**
- **Numeric - numerical data**
  - **Counts, quantities, distances, $$$, anything that can be measured…**
- **Boolean - Used to classify as True/False, Yes/No with respect to some property**
  - Garage? Breakfast included? Tall? Short? Expensive? 5-door?
- **Geographic - location or region on earth**

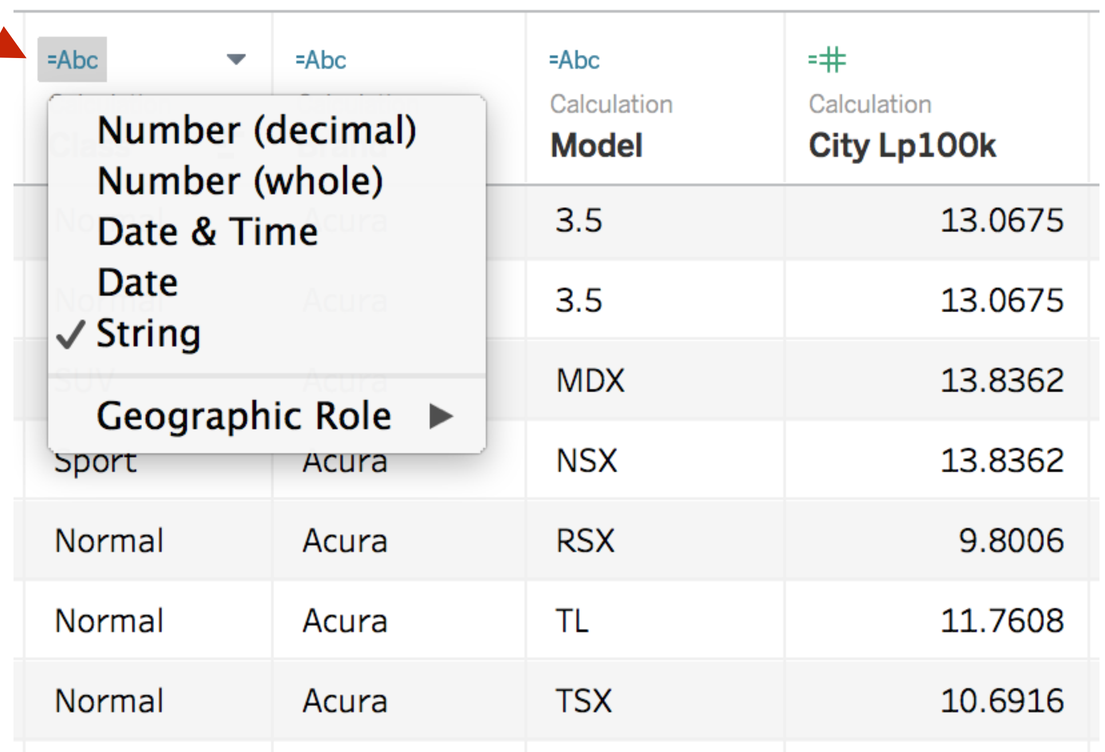
Changing Data Types

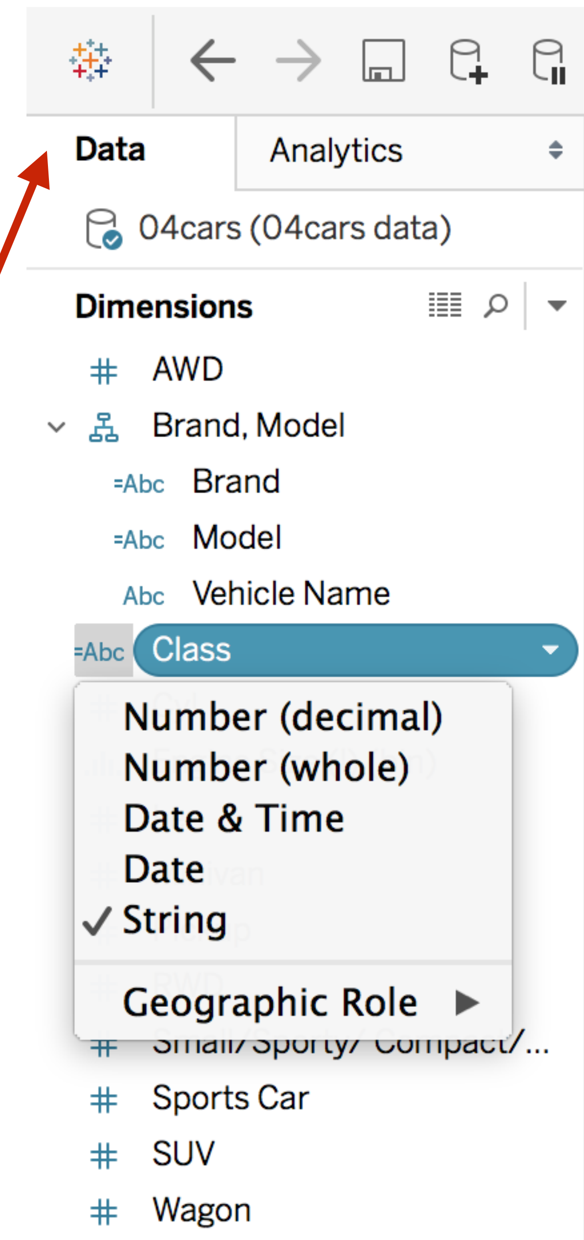
- **Changing a data type in the Data Source Page**



Changing Data Types

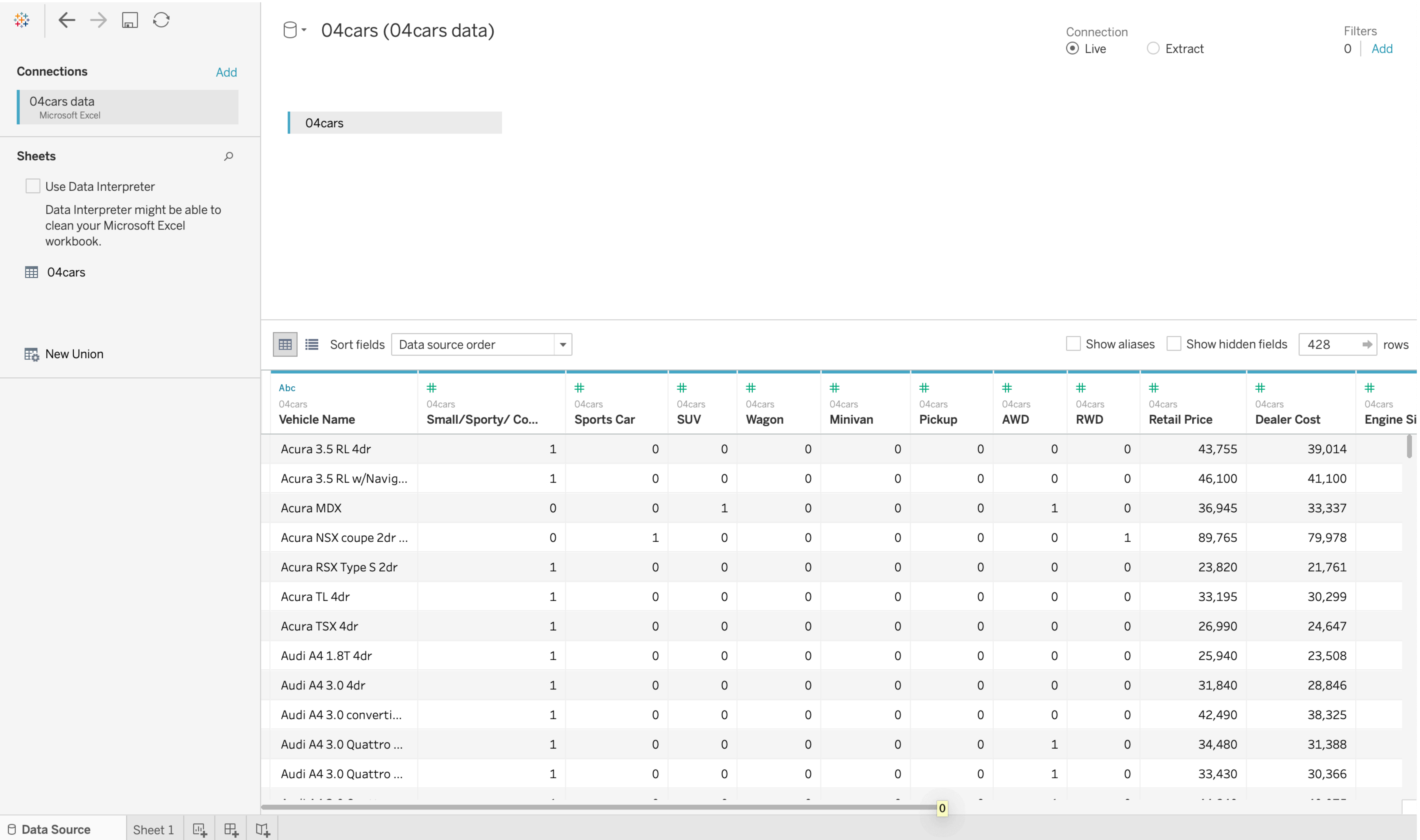
- **Changing a data type in the Data Source Page**





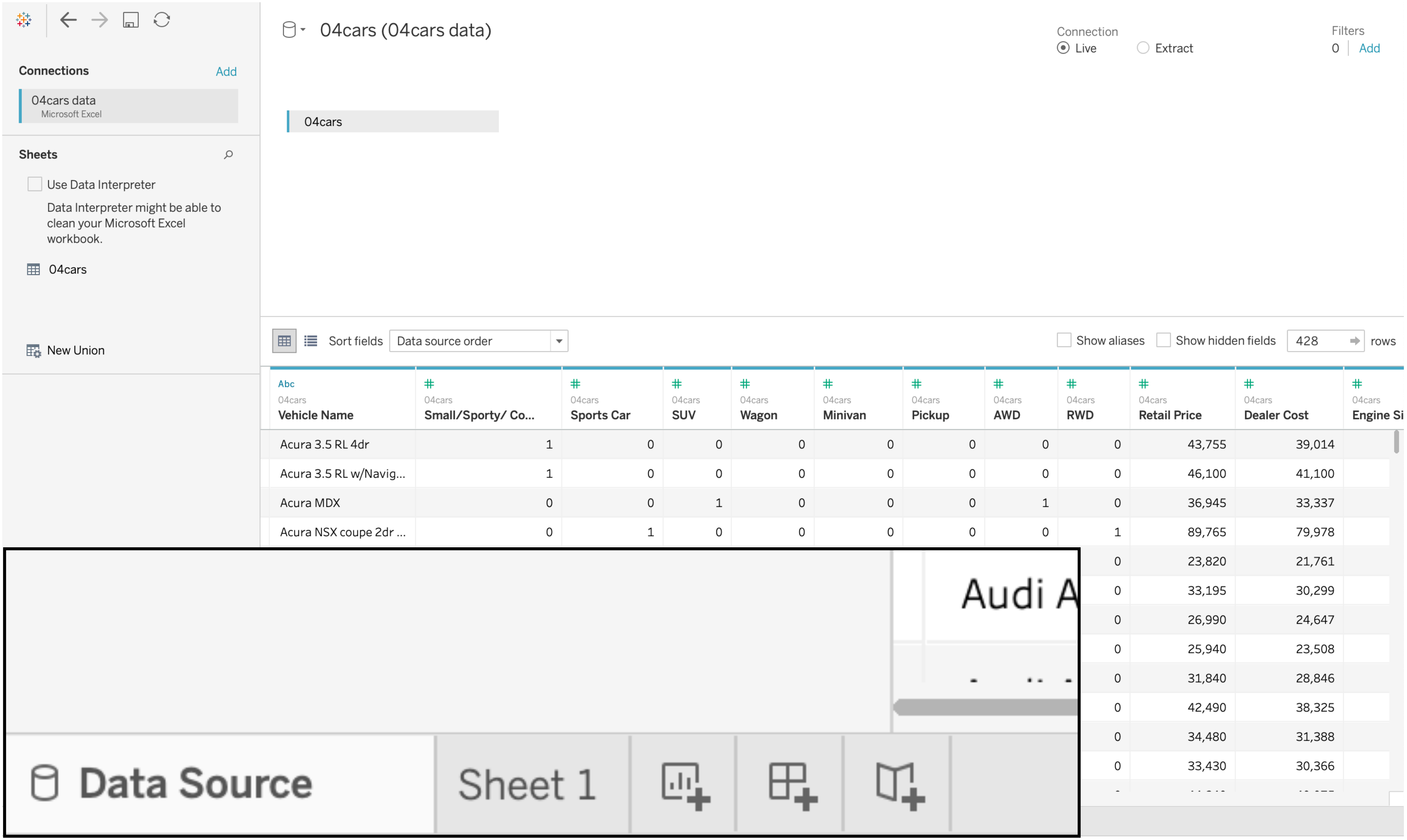
- **Changing the data type in the Data pane of a sheet**

Data source tab

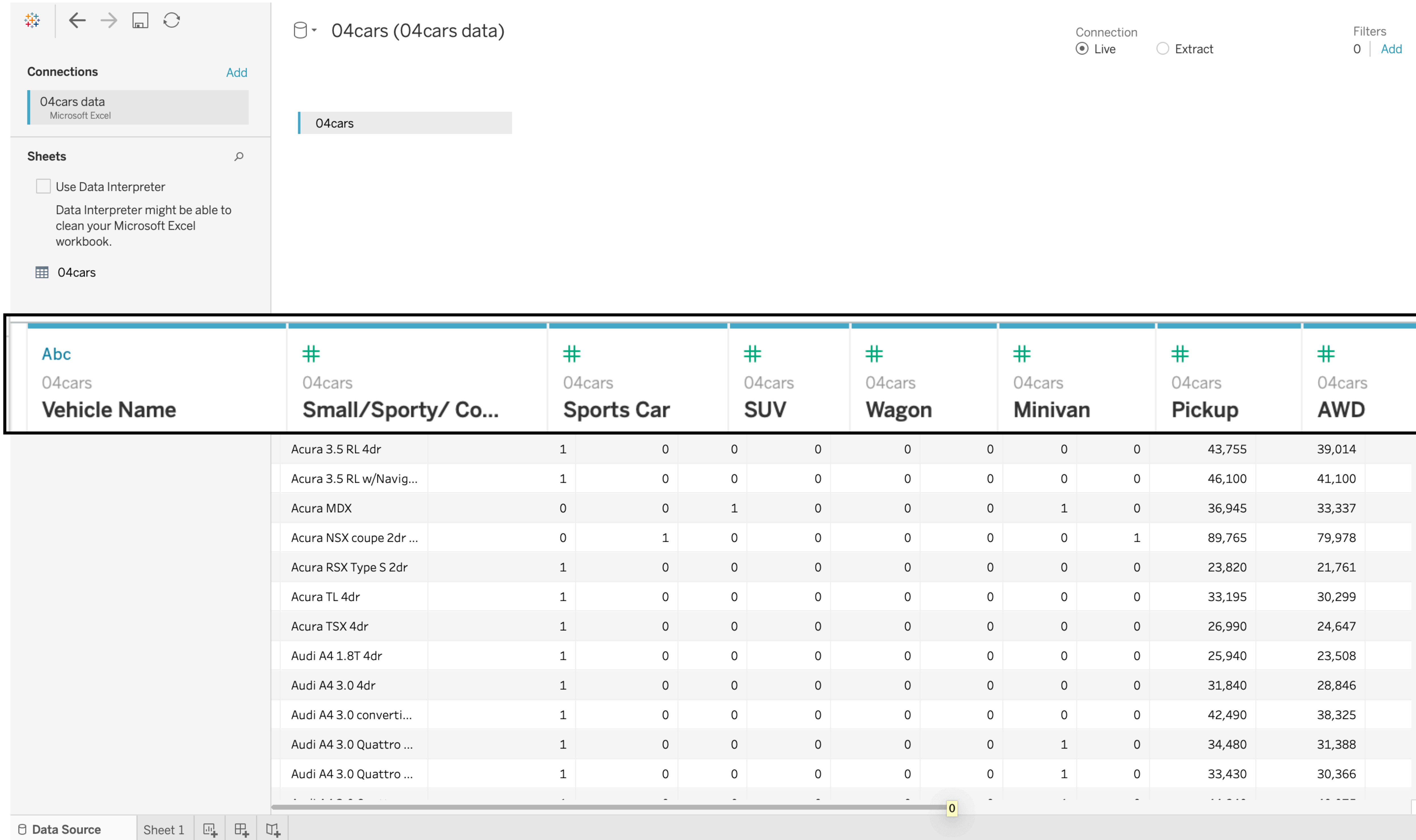


Vehicle Name	Small/Sporty/ Co...	Sports Car	SUV	Wagon	Minivan	Pickup	AWD	RWD	Retail Price	Dealer Cost	Engine Si
Acura 3.5 RL 4dr	1	0	0	0	0	0	0	0	43,755	39,014	
Acura 3.5 RL w/Navig...	1	0	0	0	0	0	0	0	46,100	41,100	
Acura MDX	0	0	1	0	0	0	1	0	36,945	33,337	
Acura NSX coupe 2dr ...	0	1	0	0	0	0	0	1	89,765	79,978	
Acura RSX Type S 2dr	1	0	0	0	0	0	0	0	23,820	21,761	
Acura TL 4dr	1	0	0	0	0	0	0	0	33,195	30,299	
Acura TSX 4dr	1	0	0	0	0	0	0	0	26,990	24,647	
Audi A4 1.8T 4dr	1	0	0	0	0	0	0	0	25,940	23,508	
Audi A4 3.0 4dr	1	0	0	0	0	0	0	0	31,840	28,846	
Audi A4 3.0 converti...	1	0	0	0	0	0	0	0	42,490	38,325	
Audi A4 3.0 Quattro ...	1	0	0	0	0	0	1	0	34,480	31,388	
Audi A4 3.0 Quattro ...	1	0	0	0	0	0	1	0	33,430	30,366	

Data source tab

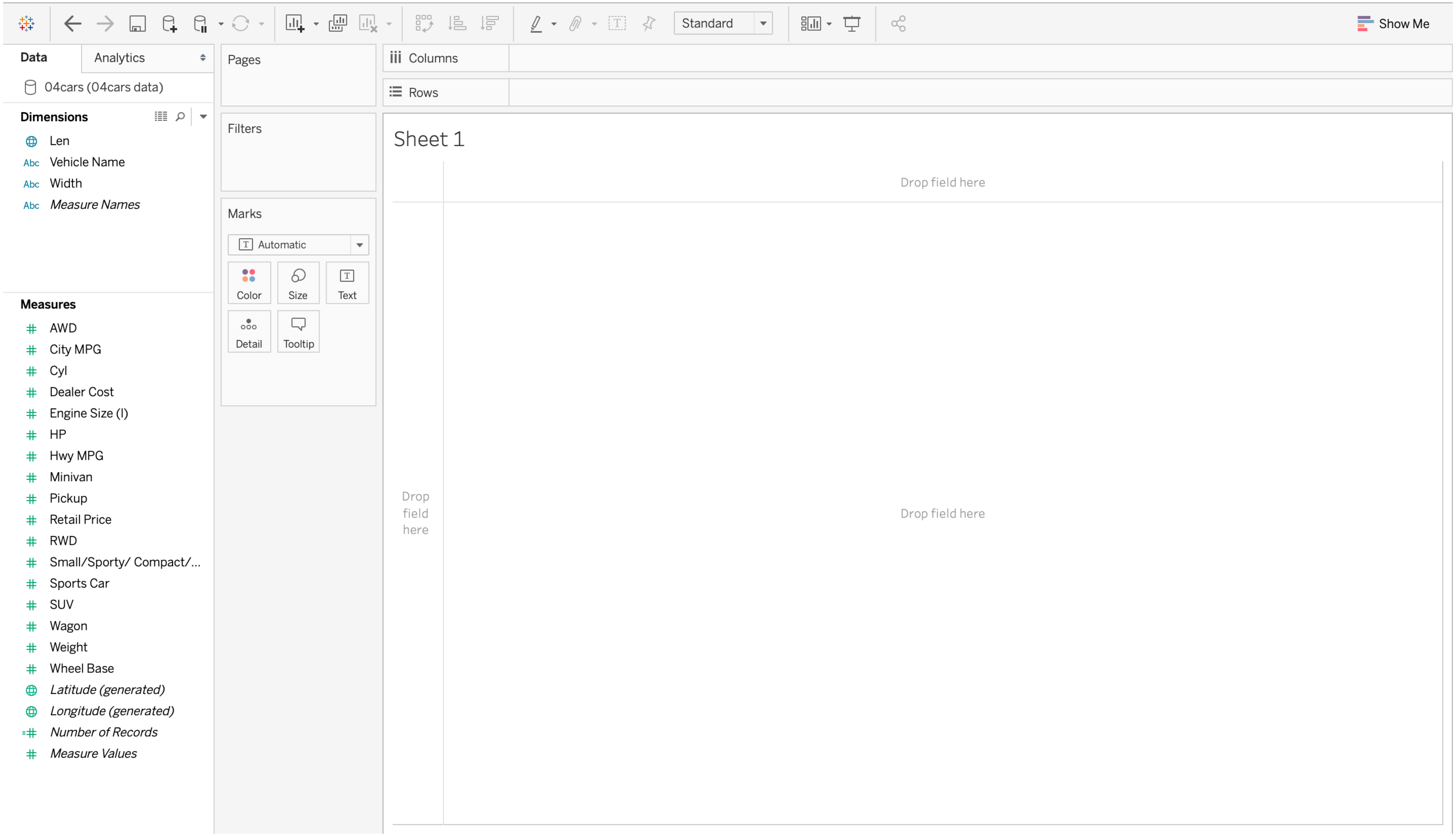


Data source tab



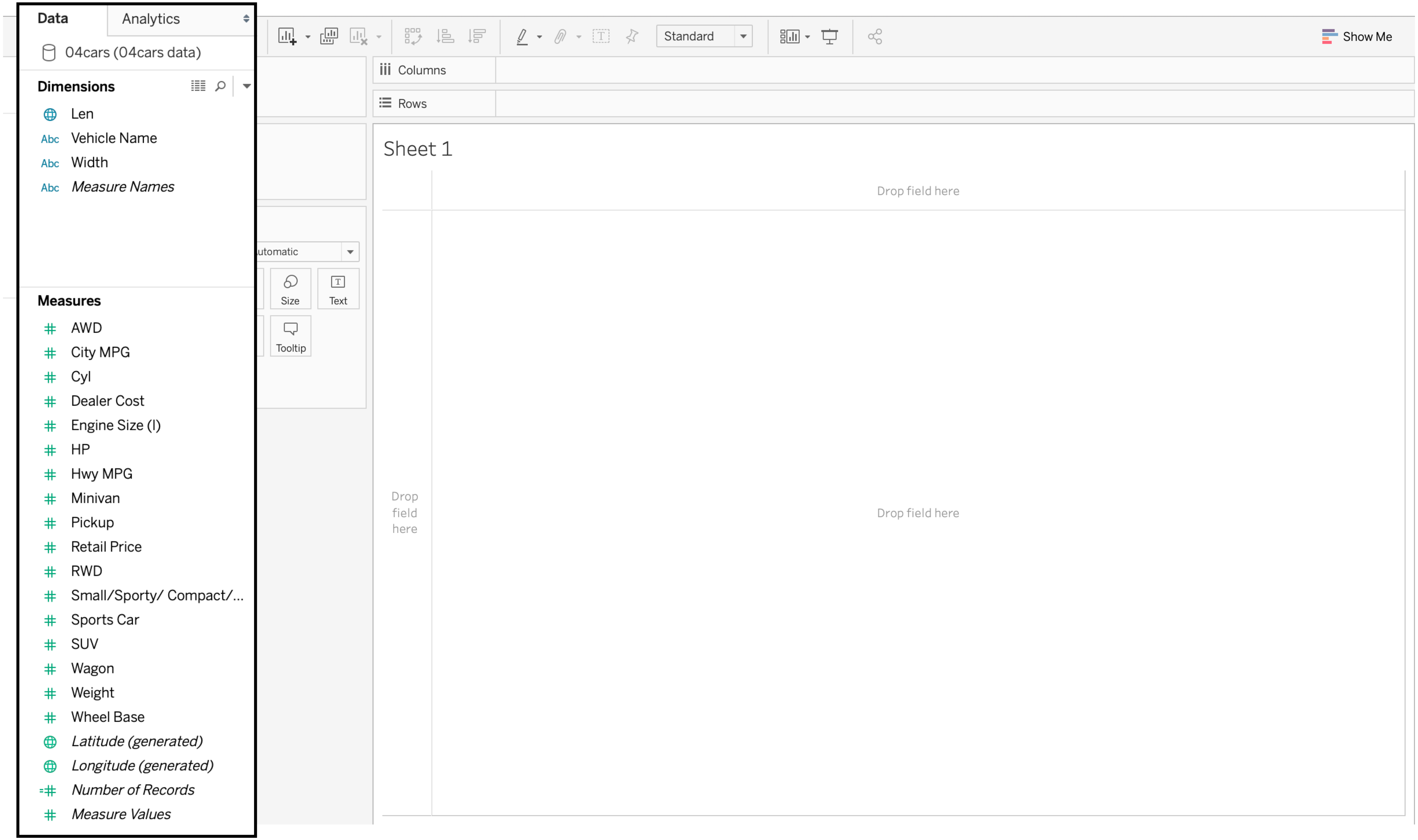
Acura 3.5 RL 4dr	1	0	0	0	0	0	0	0	43,755	39,014
Acura 3.5 RL w/Navig...	1	0	0	0	0	0	0	0	46,100	41,100
Acura MDX	0	0	1	0	0	0	1	0	36,945	33,337
Acura NSX coupe 2dr ...	0	1	0	0	0	0	0	1	89,765	79,978
Acura RSX Type S 2dr	1	0	0	0	0	0	0	0	23,820	21,761
Acura TL 4dr	1	0	0	0	0	0	0	0	33,195	30,299
Acura TSX 4dr	1	0	0	0	0	0	0	0	26,990	24,647
Audi A4 1.8T 4dr	1	0	0	0	0	0	0	0	25,940	23,508
Audi A4 3.0 4dr	1	0	0	0	0	0	0	0	31,840	28,846
Audi A4 3.0 converti...	1	0	0	0	0	0	0	0	42,490	38,325
Audi A4 3.0 Quattro ...	1	0	0	0	0	0	1	0	34,480	31,388
Audi A4 3.0 Quattro ...	1	0	0	0	0	0	1	0	33,430	30,366

View Sheet



Standard
Show Me
Data
Analytics
04cars (04cars data)
Pages
Columns
Rows
Dimensions
Len
Vehicle Name
Width
Measure Names
Filters
Marks
Automatic
Color
Size
Text
Detail
Tooltip
Sheet 1
Drop field here
Drop field here
Drop field here
Measures
AWD
City MPG
Cyl
Dealer Cost
Engine Size (l)
HP
Hwy MPG
Minivan
Pickup
Retail Price
RWD
Small/Sporty/ Compact/...
Sports Car
SUV
Wagon
Weight
Wheel Base
Latitude (generated)
Longitude (generated)
Number of Records
Measure Values

View Sheet



Data
Analytics
04cars (04cars data)
Dimensions
Len
Vehicle Name
Width
Measure Names
Measures
AWD
City MPG
Cyl
Dealer Cost
Engine Size (l)
HP
Hwy MPG
Minivan
Pickup
Retail Price
RWD
Small/Sporty/ Compact/...
Sports Car
SUV
Wagon
Weight
Wheel Base
Latitude (generated)
Longitude (generated)
Number of Records
Measure Values
Standard
Show Me
Columns
Rows
utomatic
Size
Text
Tooltip
Sheet 1
Drop field here
Drop field here
Drop field here

Changing Data Types

- **Sometimes Data Types get wrong…**
  - **Such as with the cars dataset :(**

Inferred as Geographic

Inferred as String

Tableau guessed wrong because of the values encountered in the dataset

🌐 04cars Len	Abc 04cars Width
153	66
183	69
183	68
183	69
*	*
180	74

Changing Data Types

- **Sometimes Data Types get wrong…**
  - **Such as with the cars dataset :(**

Inferred as Geographic

Inferred as String

04cars Len	Abc 04cars Width
153	66
183	69
183	68
183	69
*	*
180	74

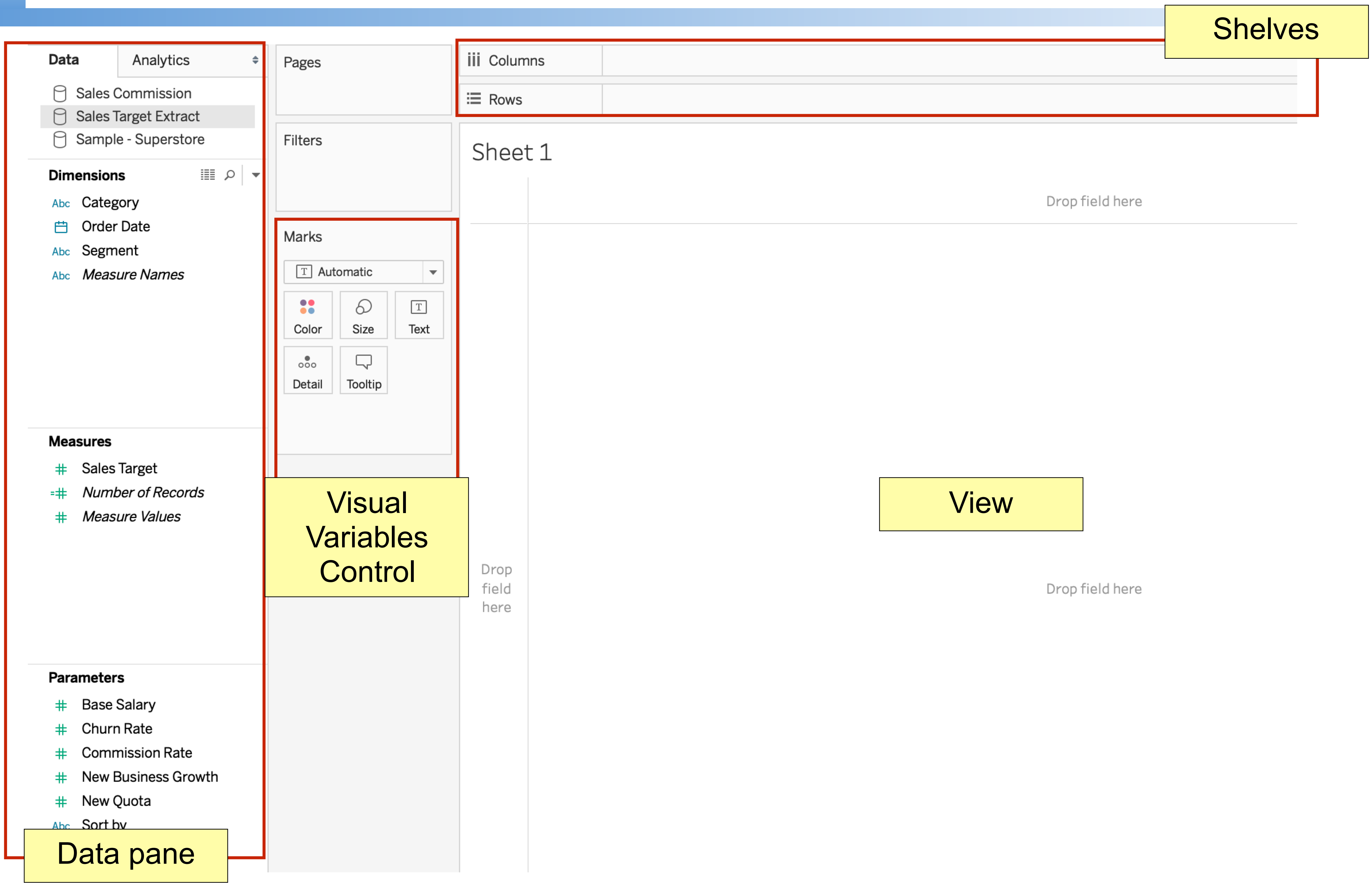
Tableau guessed wrong because of the values encountered in the dataset

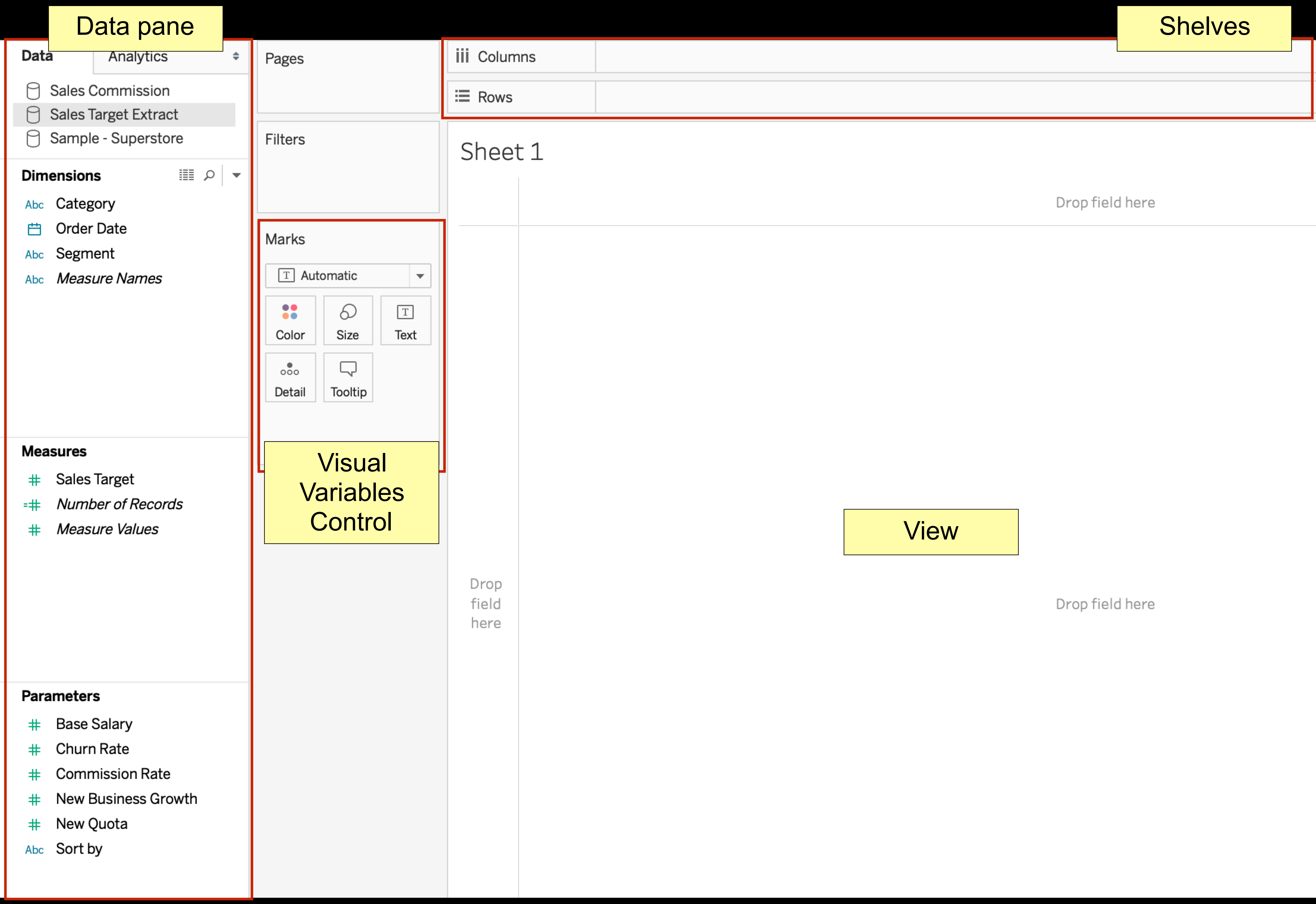
- **Action required…**

Data analyst must correct them !!!

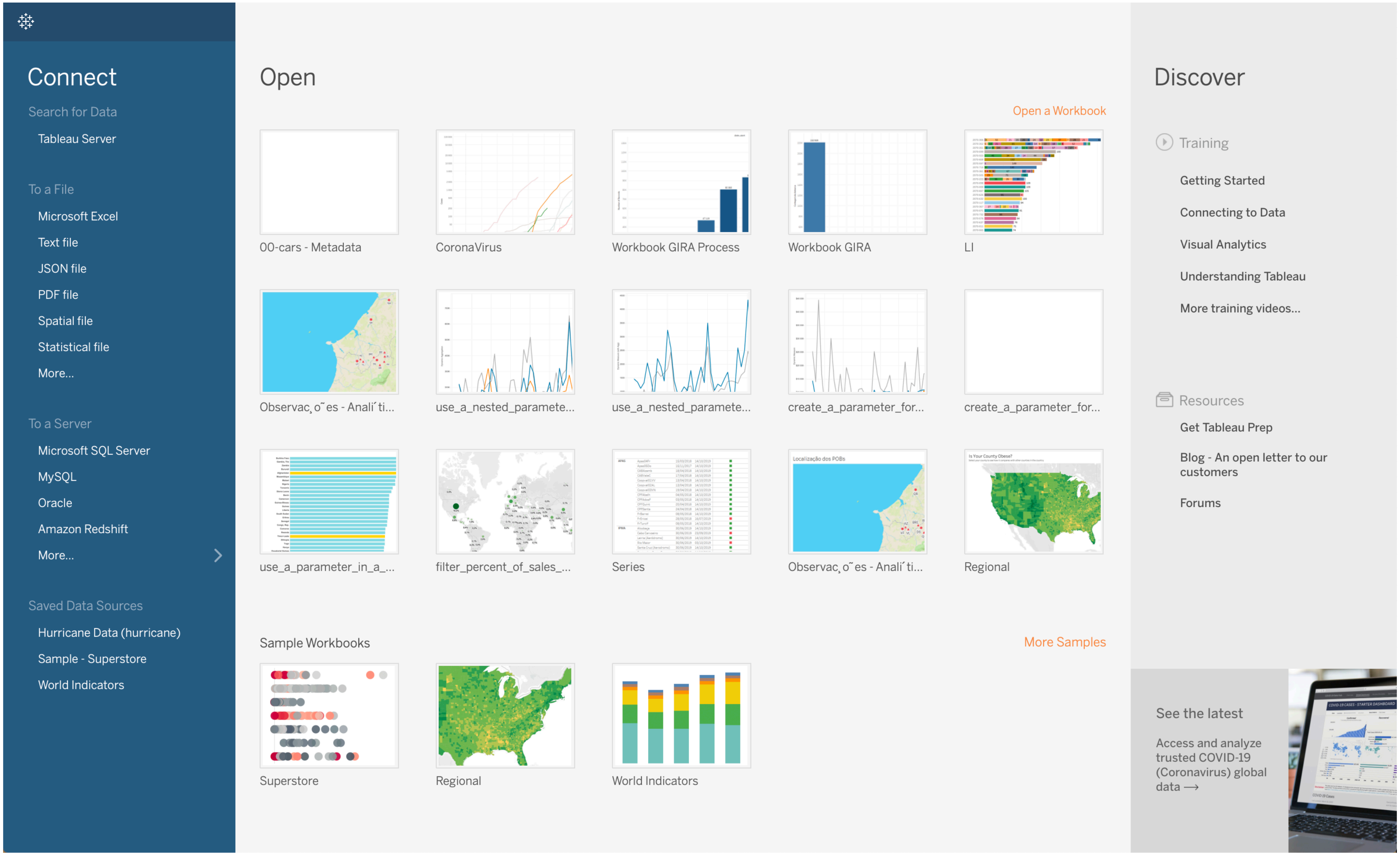
# 04cars Len	# 04cars Width
153	66
183	69
183	68
183	69
null	*null*
180	74

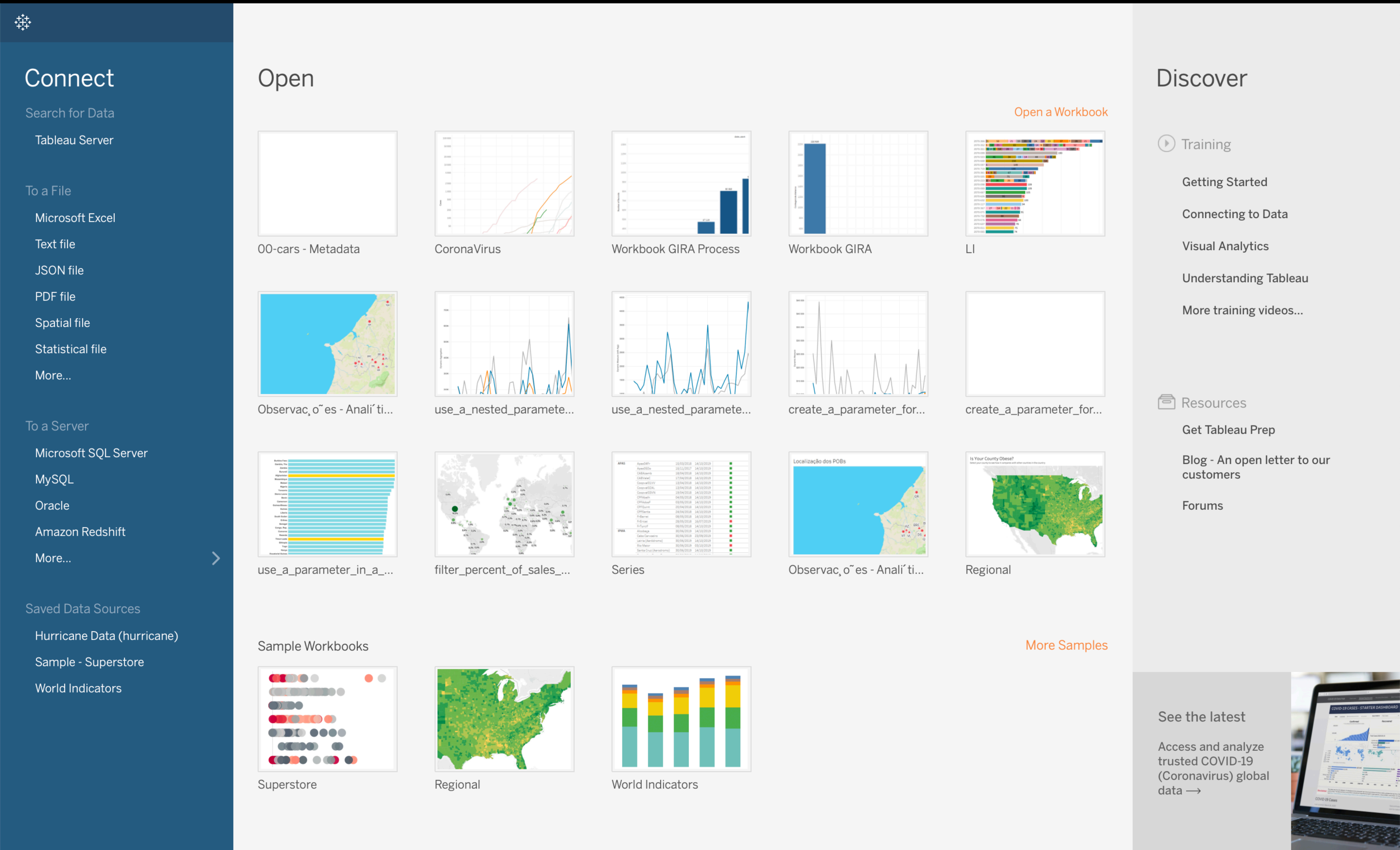
Tableau Sheet View

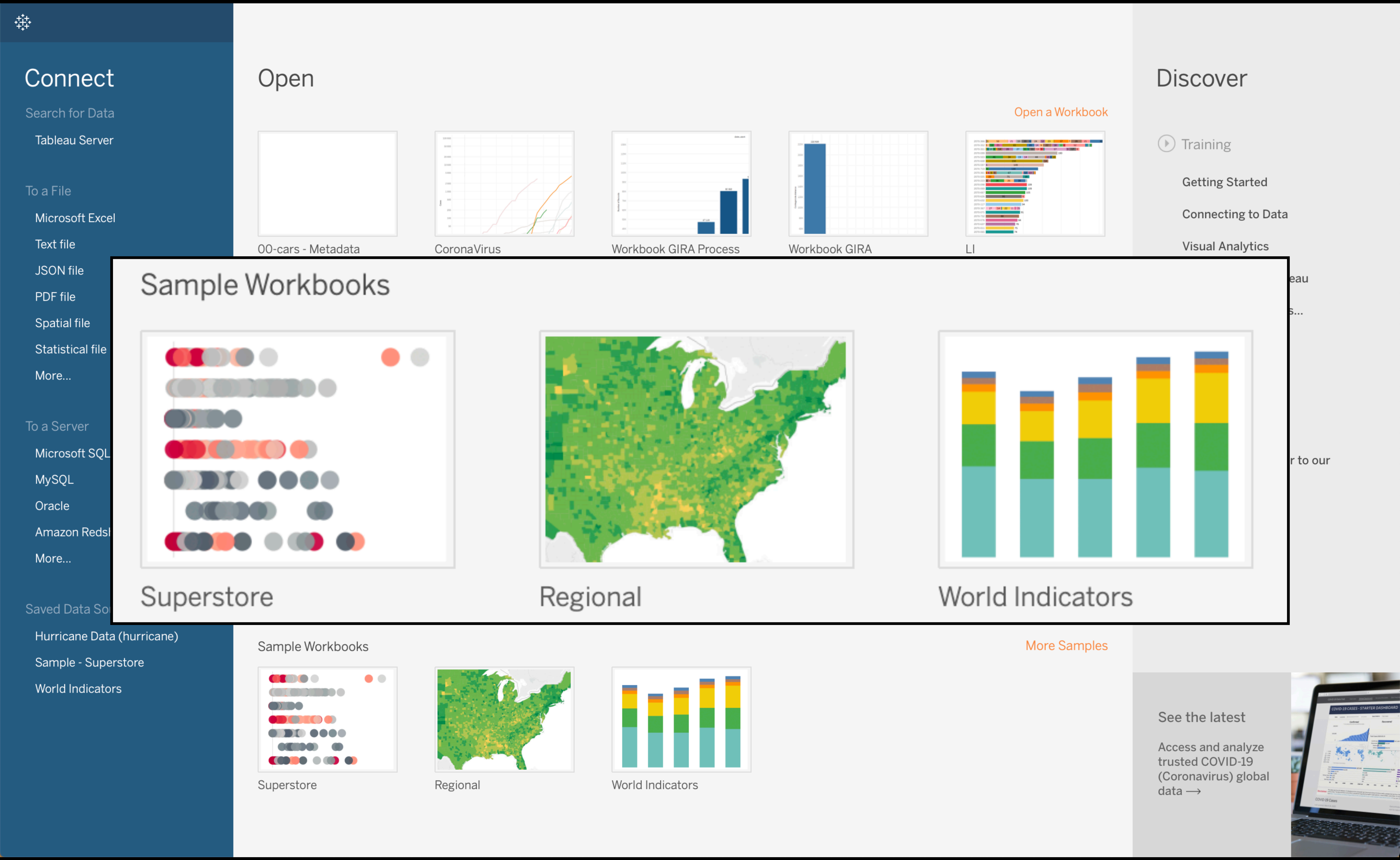



Data pane
Shelves
Data
Analytics
Sales Commission
Sales Target Extract
Sample - Superstore
Dimensions
Category
Order Date
Segment
Measure Names
Measures
Sales Target
Number of Records
Measure Values
Parameters
Base Salary
Churn Rate
Commission Rate
New Business Growth
New Quota
Sort by
Pages
Filters
Marks
Automatic
Color
Size
Text
Detail
Tooltip
Visual Variables Control
Columns
Rows
Sheet 1
Drop field here
Drop field here
Drop field here
View

Use the samples to play with




Connect
Search for Data
Tableau Server
To a File
Microsoft Excel
Text file
JSON file
PDF file
Spatial file
Statistical file
More...
To a Server
Microsoft SQL Server
MySQL
Oracle
Amazon Redshift
More...
Saved Data Sources
Hurricane Data (hurricane)
Sample - Superstore
World Indicators
Open
Open a Workbook
00-cars - Metadata
CoronaVirus
Workbook GIRA Process
Workbook GIRA
LI
Observac¸o˜es - Anali´ti...
use_a_nested_paramete...
use_a_nested_paramete...
create_a_parameter_for...
create_a_parameter_for...
use_a_parameter_in_a_...
filter_percent_of_sales_...
Series
Observac¸o˜es - Anali´ti...
Regional
Sample Workbooks
More Samples
Superstore
Regional
World Indicators
Discover
Training
Getting Started
Connecting to Data
Visual Analytics
Understanding Tableau
More training videos...
Resources
Get Tableau Prep
Blog - An open letter to our customers
Forums
See the latest
Access and analyze trusted COVID-19 (Coronavirus) global data ⟶


Connect
Search for Data
Tableau Server
To a File
Microsoft Excel
Text file
JSON file
PDF file
Spatial file
Statistical file
More...
To a Server
Microsoft SQL
MySQL
Oracle
Amazon Redsl
More...
Saved Data So
Hurricane Data (hurricane)
Sample - Superstore
World Indicators
Open
Open a Workbook
00-cars - Metadata
CoronaVirus
Workbook GIRA Process
Workbook GIRA
LI
Sample Workbooks
Superstore
Regional
World Indicators
Sample Workbooks
More Samples
Superstore
Regional
World Indicators
Discover
Training
Getting Started
Connecting to Data
Visual Analytics
eau
s...
r to our
See the latest
Access and analyze trusted COVID-19 (Coronavirus) global data ⟶

Dimensions and Measures

Dimensions and Measures

- **Independent vs. Dependent Variables**
- **Dimensions and Measures**
- **Discrete vs. Continuous**
- **Examples**
- **Data Inspection**
- **Filters (continuous versus discrete)**
- **Color (continuous versus discrete)**

Dimensions and Measures

- **Independent vs. Dependent Variables**
- **Dimensions and Measures**
- **Discrete vs. Continuous**
- **Examples**
- **Data Inspection**
- **Filters (continuous versus discrete)**
- **Color (continuous versus discrete)**

Independent vs. Dependent Variables

Independent vs. Dependent Variables

- **When we analyze data we usually sort the variables in two different groups:**
  - **Independent variables**
    - Independent variables provide **context/structure** to our numerical data values
    - Usually correspond to categorical data (months, years, region, state, country, …) or discrete numbers

Independent vs. Dependent Variables

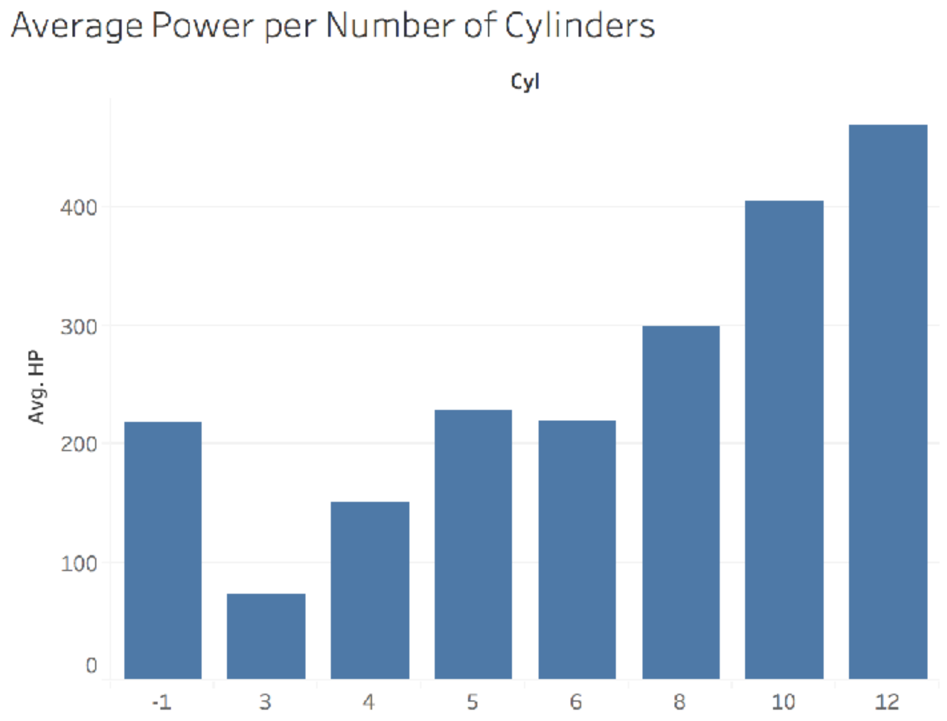
- **When we analyze data we usually sort the variables in two different groups:**
  - **Independent variables**
    - Independent variables provide **context/structure** to our numerical data values
    - Usually correspond to categorical data (months, years, region, state, country, …) or discrete numbers
  - **Dependent variables**
    - Their values alone do not have any meaning requiring some kind of context associated
    - Normally they are continuous or discrete figures (revenue, investment, weight, …)

Independent vs. Dependent Variables

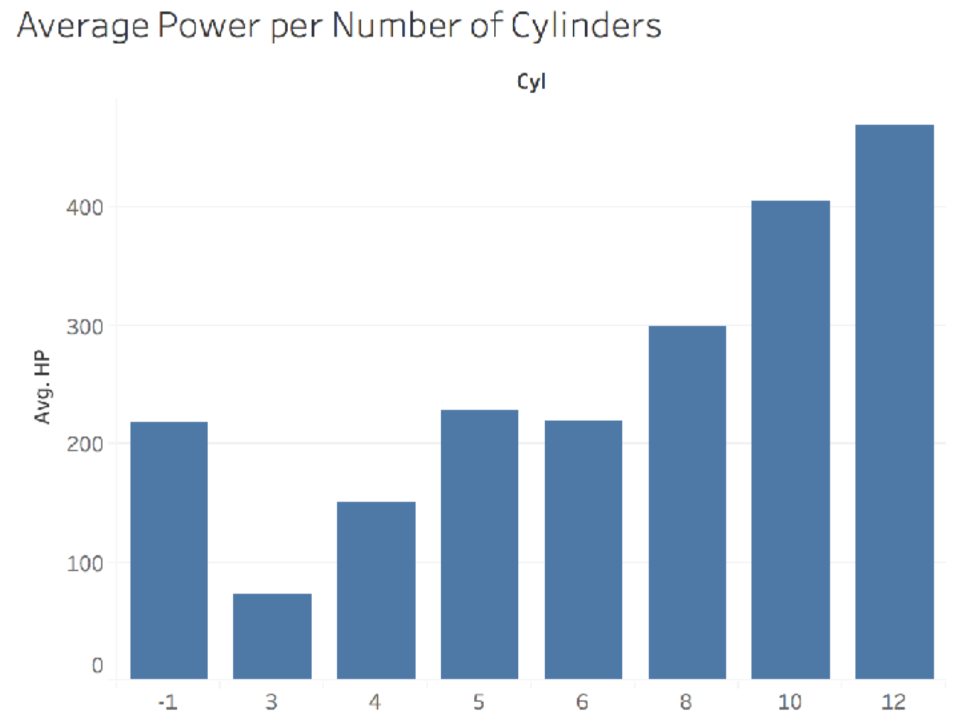
- **When we analyze data we usually sort the variables in two different groups:**
  - **Independent variables**
    - Independent variables provide **context/structure** to our numerical data values
    - Usually correspond to categorical data (months, years, region, state, country, …) or discrete numbers
  - **Dependent variables**
    - Their values alone do not have any meaning requiring some kind of context associated
    - Normally they are continuous or discrete figures (revenue, investment, weight, …)
  - **When looking at our values, we read them in a context:**
    - Revenue per month, investment per region, average cargo weight, …
    - Dependent variables are a function of the independent variables

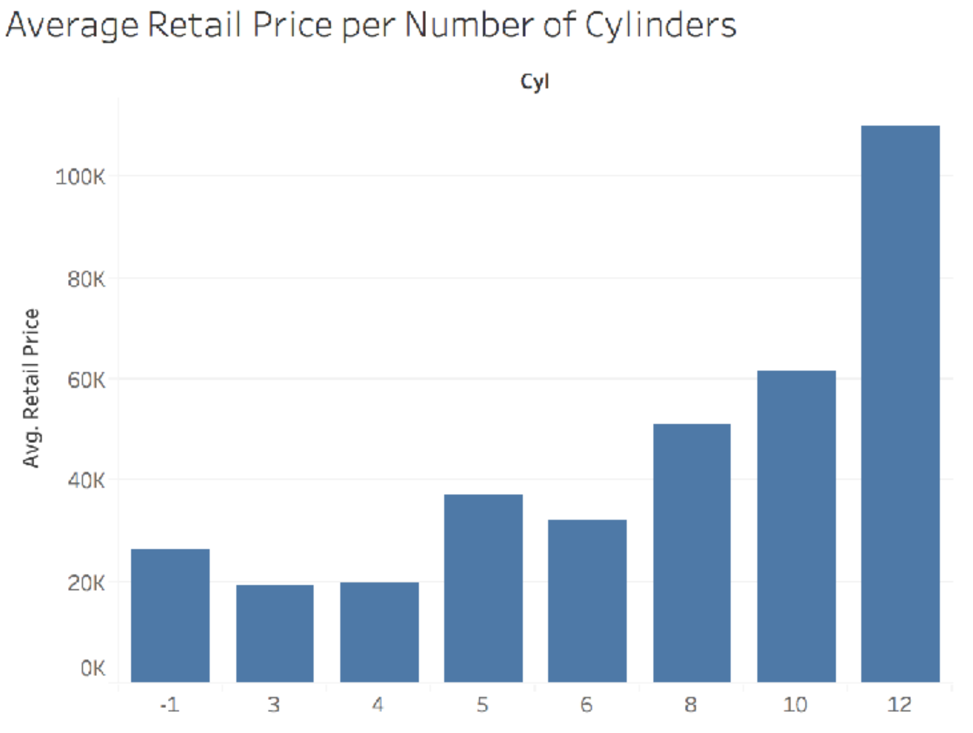
Independent vs. Dependent Variables

Independent vs. Dependent Variables

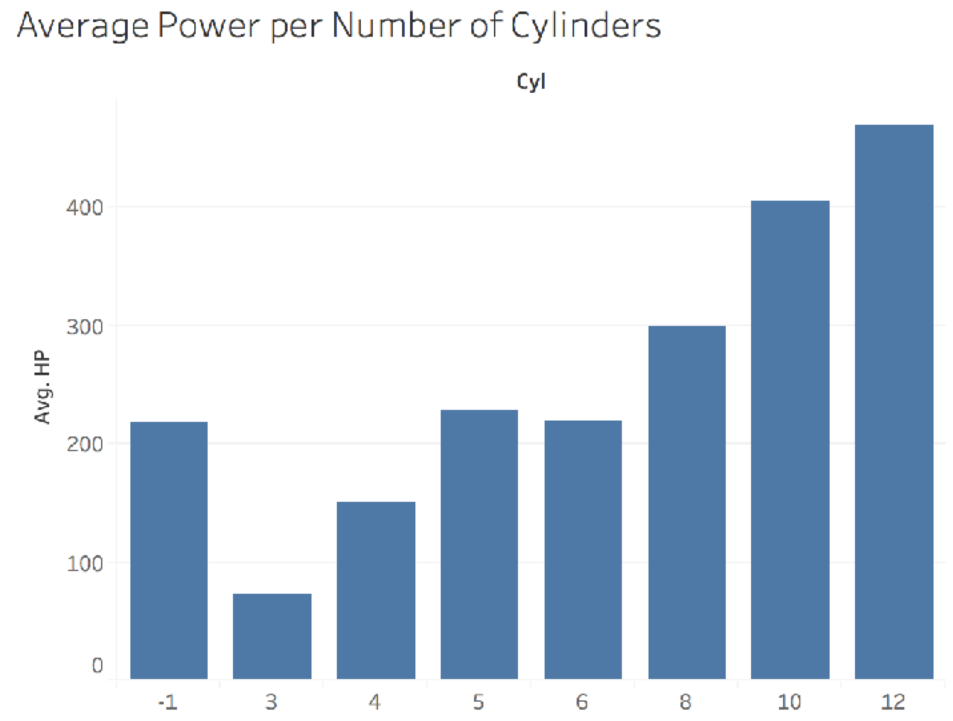


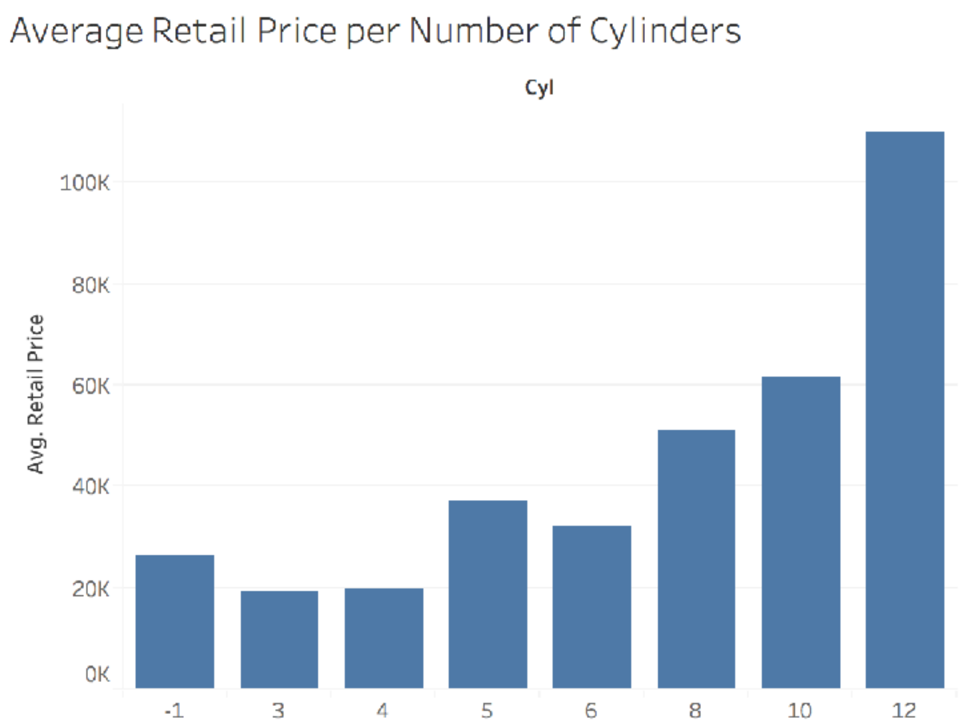
Independent vs. Dependent Variables

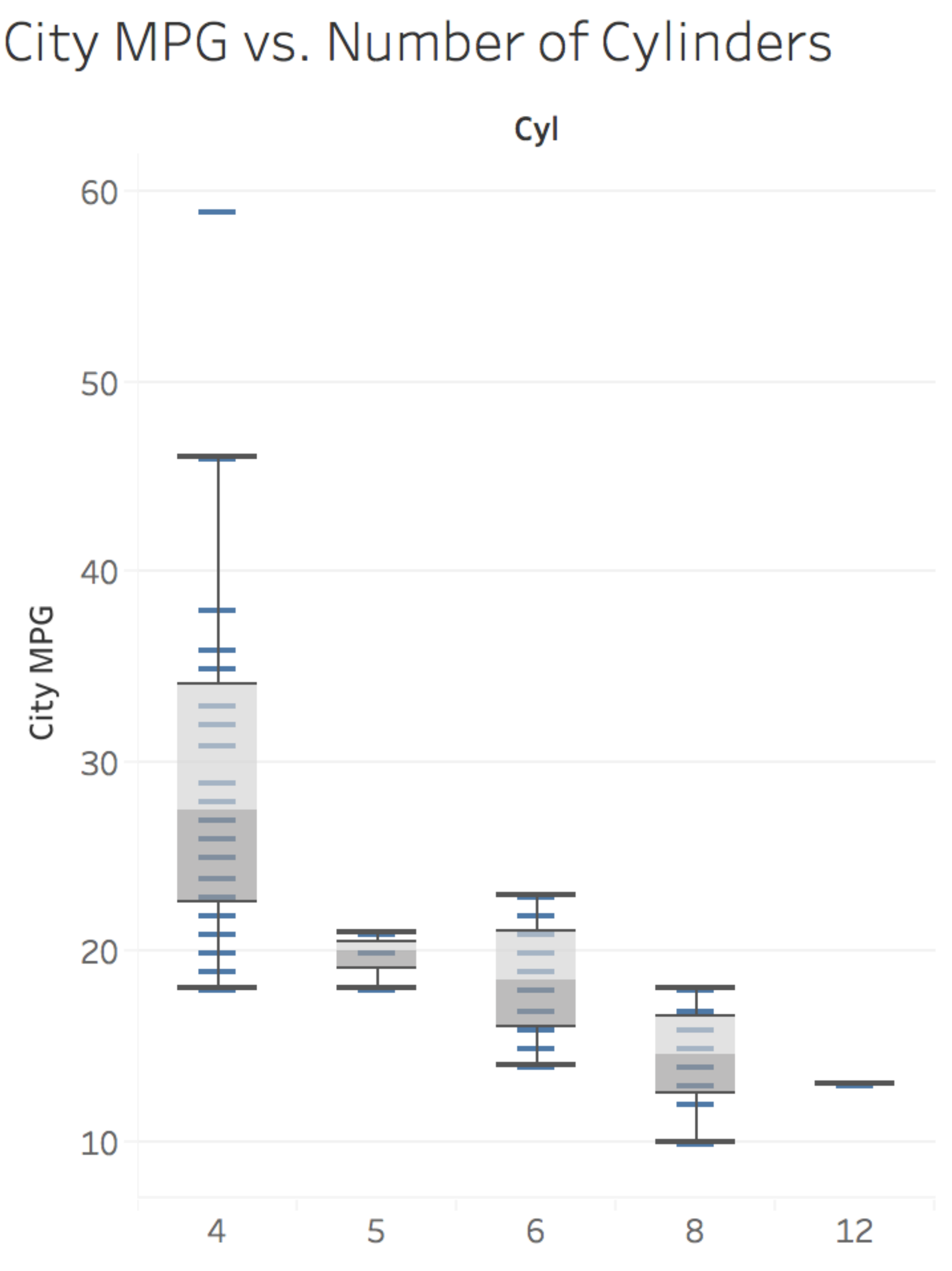




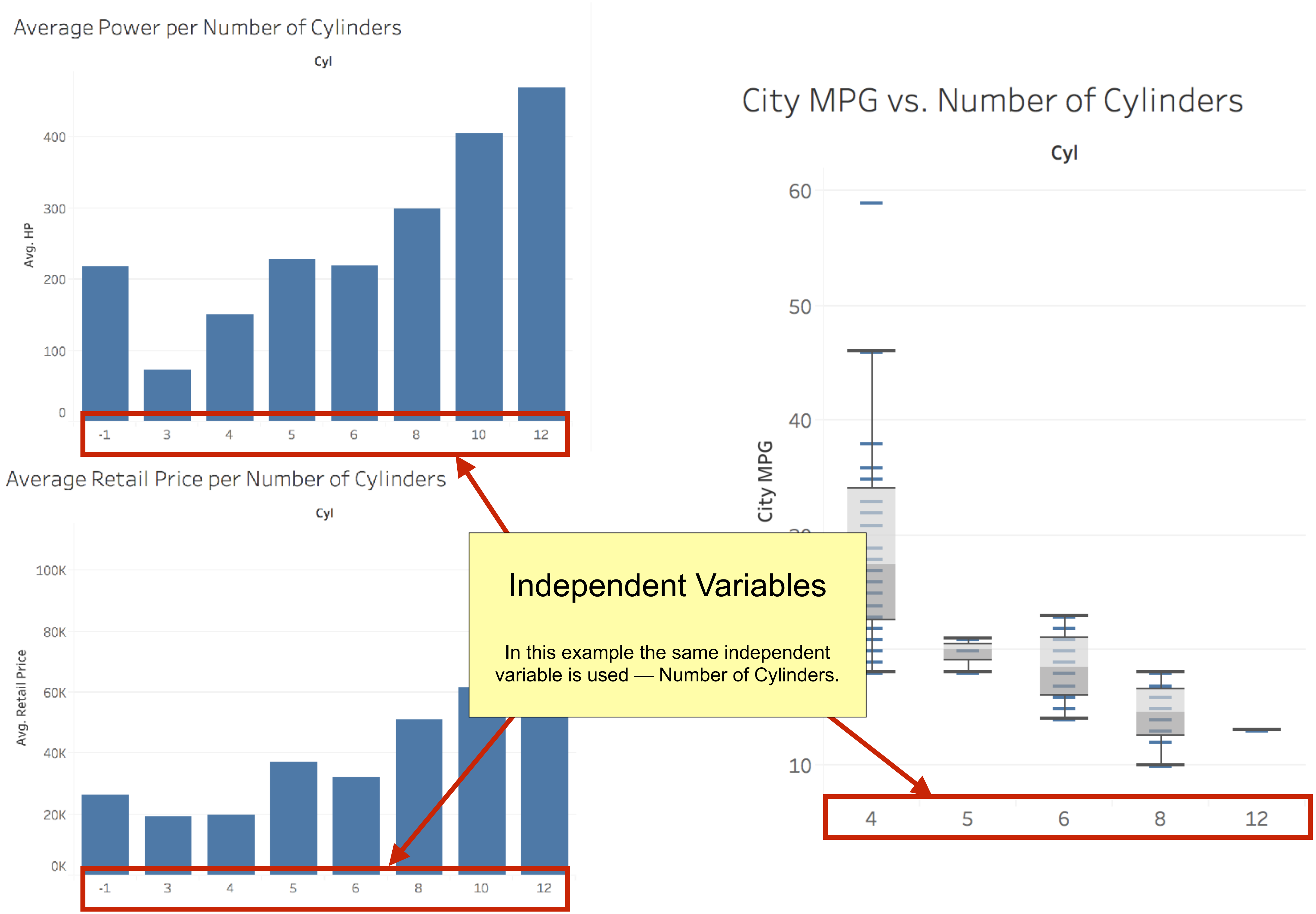
Independent vs. Dependent Variables



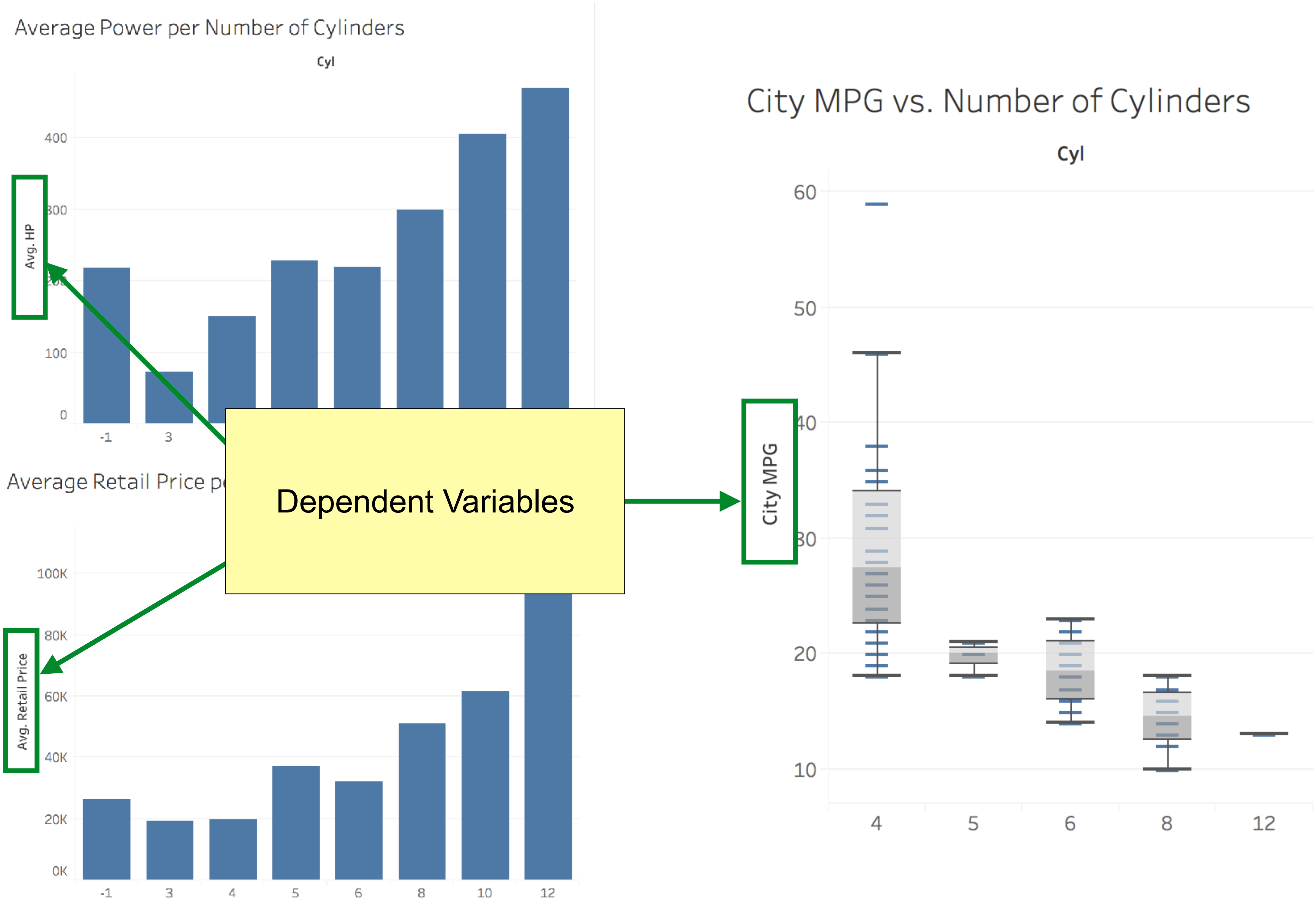




Independent vs. Dependent Variables



Independent vs. Dependent Variables



Dimensions and Measures

Dimensions and Measures

- **Dimensions and Measures are terms from Data Warehousing and Multidimensional Models**

Dimensions and Measures

- **Dimensions** and **Measures** are terms from Data Warehousing and Multidimensional Models
- **Dimensions** allow data analysis from various perspectives
  - ***Time*: breakdown sales per year, quarter, month, etc.**
  - ***Product*: which product bring the most revenue**
  - ***Supplier*: who are the ones that deliver goods in time**

Dimensions and Measures

- **Dimensions and Measures are terms from Data Warehousing and Multidimensional Models**
- **Dimensions allow data analysis from various perspectives**
  - ***Time*: breakdown sales per year, quarter, month, etc.**
  - ***Product*: which product bring the most revenue**
  - ***Supplier*: who are the ones that deliver goods in time**
- **Measures are numeric representations of facts that occurred**
  - **Sales amount**
  - **Store percentage of profit**
  - **Number of returned products**

Dimensions and Measures

- **Dimensions** and **Measures** are terms from Data Warehousing and Multidimensional Models
- **Dimensions** allow data analysis from various perspectives
  - ***Time*: breakdown sales per year, quarter, month, etc.**
  - ***Product*: which product bring the most revenue**
  - ***Supplier*: who are the ones that deliver goods in time**

Independent Variables

Answer questions like:

Who? What? When? Where?

- **Measures** are numeric representations of facts that occurred
  - **Sales amount**
  - **Store percentage of profit**
  - **Number of returned products**

Dimensions and Measures

- **Dimensions and Measures are terms from Data Warehousing and Multidimensional Models**
- **Dimensions allow data analysis from various perspectives**
  - ***Time*: breakdown sales per year, quarter, month, etc.**
  - ***Product*: which product bring the most revenue**
  - ***Supplier*: who are the ones that deliver goods in time**

Independent Variables

Answer questions like:

Who? What? When? Where?

- **Measures are numeric representations of facts that occurred**
  - **Sales amount**
  - **Store percentage of profit**
  - **Number of returned products**

Dependent Variables

Dimensions and Measures

- **Dimensions and Measures are terms from Data Warehousing and Multidimensional Models**
- **Dimensions allow data analysis from various perspectives**
  - ***Time*: breakdown sales per year, quarter, month, etc.**
  - ***Product*: which product bring the most revenue**
  - ***Supplier*: who are the ones that deliver goods in time**

Independent Variables

Answer questions like:

Who? What? When? Where?

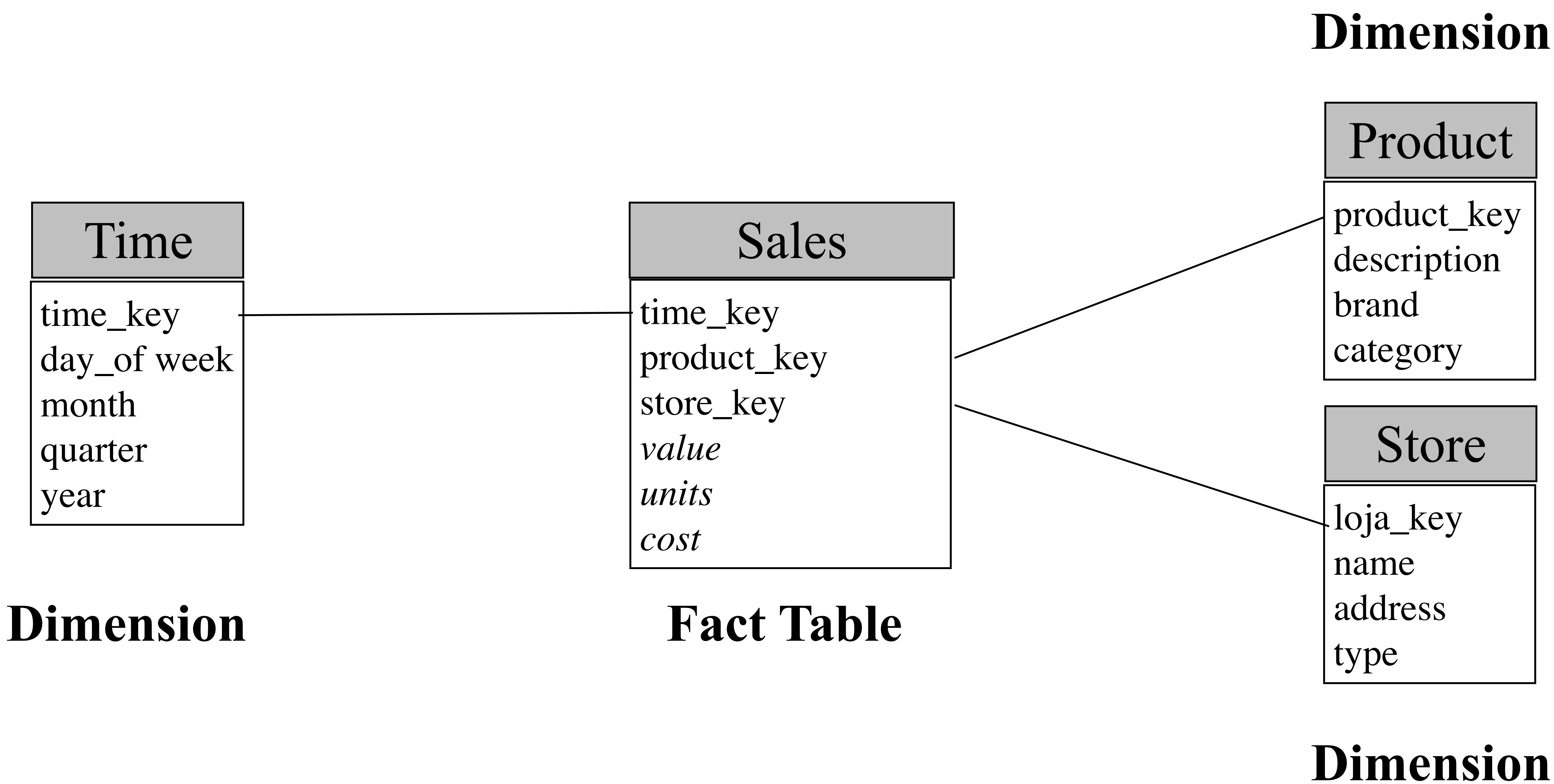
- **Measures are numeric representations of facts that occurred**
  - **Sales amount**
  - **Store percentage of profit**
  - **Number of returned products**

Dependent Variables

Aggregated Values

Multidimensional Model: Star Schema

- **Data for the first quarter for all stores by brand**



Typical result

- **Data for the first quarter for all stores by brand**

Brand	Dollar amount sold	Sold Units
M-1	780	263
M-2	1044	509
M-3	213	444
M-4	95	39

Distinct values for the selected attribute

Typical result

- **Data for the first quarter for all stores by brand**

Brand	Dollar amount sold	Sold Units
M-1	780	263
M-2	1044	509
M-3	213	444
M-4	95	39

Metrics

Distinct values for the selected attribute

Typical result

- **Data for the first quarter for all stores by brand**

Brand	Dollar amount sold	Sold Units
M-1	780	263
M-2	1044	509
M-3	213	444
M-4	95	39

Metrics

Distinct values for the selected attribute

Textual Attribute of a Dimension

Typical SQL query for StarSchema

- **Data for the first quarter for all stores by brand**

```
select p.brand, sum(f.value), sum(f.units)
from sales f, product p, time t

where f.product_key = p.product_key
        and f.time_key = t.time_key
        and f.quarter = “Q1 1996”

group by p.brand
order by p.brand
```

Typical SQL query for StarSchema

- **Data for the first quarter for all stores by brand**

Selecting the columns

```
select p.brand, sum(f.value), sum(f.units)
from sales f, product p, time t

where f.product_key = p.product_key
        and f.time_key = t.time_key
        and f.quarter = “Q1 1996”

group by p.brand
order by p.brand
```

Typical SQL query for StarSchema

- **Data for the first quarter for all stores by brand**

Selecting the columns

select p.brand, **sum**(f.value), **sum**(f.units) ← Aggregation
from sales f, product p, time t

where f.product_key = p.product_key
and f.time_key = t.time_key
and f.quarter = “Q1 1996”

group by p.brand
order by p.brand

FACULDADE DE CIÊNCIAS E TECNOLOGIA UNIVERSIDADE NOVA DE LISBOA

Typical SQL query for StarSchema

- **Data for the first quarter for all stores by brand**

Selecting the columns

select p.brand, **sum**(f.value), **sum**(f.units) ← Aggregation

from sales f, product p, time t ← aliases

where f.product_key = p.product_key
 and f.time_key = t.time_key
 and f.quarter = "Q1 1996"

group by p.brand
order by p.brand

FACULDADE DE CIÊNCIAS E TECNOLOGIA UNIVERSIDADE NOVA DE LISBOA

Typical SQL query for StarSchema

- **Data for the first quarter for all stores by brand**

```
select p.brand, sum(f.value), sum(f.units)   <-- Aggregation   (Selecting the columns)
from sales f, product p, time t              <-- aliases

where f.product_key = p.product_key          <-- Join constraint
      and f.time_key = t.time_key            <-- Join constraint
      and f.quarter = “Q1 1996”

group by p.brand
order by p.brand
```

Typical SQL query for StarSchema

- **Data for the first quarter for all stores by brand**

Selecting the columns

select p.brand, **sum**(f.value), **sum**(f.units) ← Aggregation

from sales f, product p, time t ← aliases

where f.product_key = p.product_key ← Join constraint

and f.time_key = t.time_key ← Join constraint

and f.quarter = "Q1 1996" ← Application constraint

group by p.brand

order by p.brand

FACULDADE DE CIÊNCIAS E TECNOLOGIA UNIVERSIDADE NOVA DE LISBOA

Typical SQL query for StarSchema

- **Data for the first quarter for all stores by brand**

Selecting the columns

```
select p.brand, sum(f.value), sum(f.units)      <- Aggregation
from sales f, product p, time t                 <- aliases

where f.product_key = p.product_key             <- Join constraint
      and f.time_key = t.time_key               <- Join constraint
      and f.quarter = "Q1 1996"                 <- Application constraint

group by p.brand                                <- Grouping
order by p.brand
```

Typical SQL query for StarSchema

- **Data for the first quarter for all stores by brand**

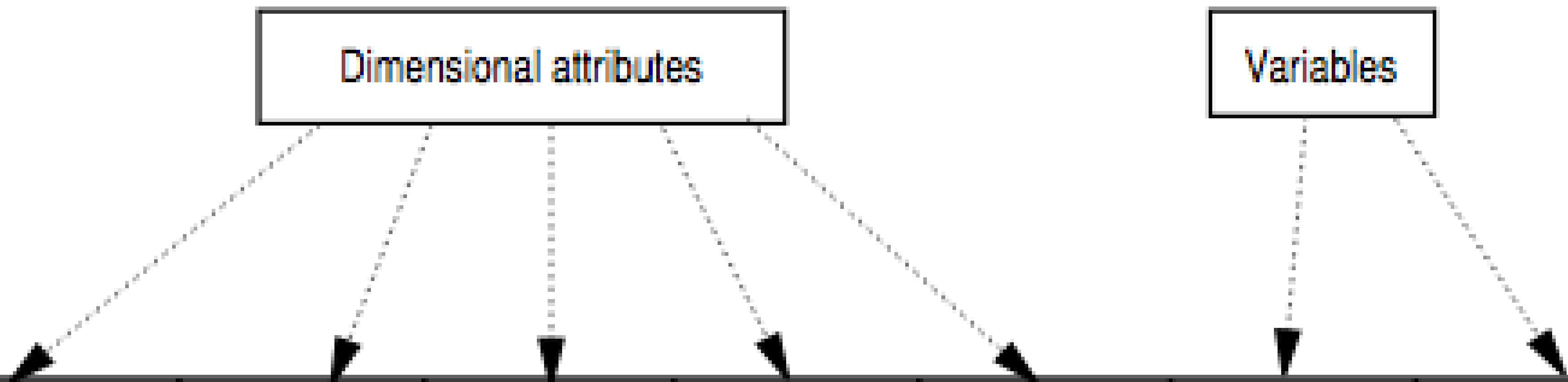
Selecting the columns

select p.brand, **sum**(f.value), **sum**(f.units) ← Aggregation
from sales f, product p, time t ← aliases

where f.product_key = p.product_key ← Join constraint
and f.time_key = t.time_key ← Join constraint
and f.quarter = "Q1 1996" ← Application constraint

group by p.brand ← Grouping
order by p.brand ← Sorting

From a rowset to an analytical view



Time	Product	Scenario	Store	Customer Type	# Units	Sales
2	10	1	12	3	2	$1500
2	11	1	12	3	3	$2250
2	12	1	12	3	2	$1500

Classical OLAP view

Store.Paris

	Actual				Plan			
	Toys		Clothes		Toys		Clothes	
	Sales	Costs	Sales	Costs	Sales	Costs	Sales	Costs
Q1	320	200	825	750	525	603	750	629
Q2	225	220	390	250	554	600	365	400
Q3	700	600	425	630	653	725	720	530
Q4	880	850	875	700	893	875	890	889

Inefficient OLAP view

				Q1	Q2	Q3
Actual	Paris	Toys	Sales	320	225	700
			Costs	200	220	600
		Clothes	Sales	825	390	425
			Costs	750	250	630
	NYC	Toys	Sales	500	310	880
			Costs	450	500	850
		Clothes	Sales	210	625	875
			Costs	225	600	700
Plan	Paris	Toys	Sales	525	554	653
			Costs	603	600	725
		Clothes	Sales	750	365	320
			Costs	629	400	530
	NYC	Toys	Sales	460	520	810
			Costs	325	610	875
		Clothes	Sales	655	725	890
			Costs	780	650	889

Dimensions and Measures in Tableau

- **Depending on the contents, Tableau initially assigns each field in the data source to either:**
  - **Dimensions or**
  - **Measures**

Dimensions and Measures in Tableau

- **Depending on the contents, Tableau initially assigns each field in the data source to either:**
  - **Dimensions or**
  - **Measures**
- **Fields with categorical data such as names, dates, or geographical data are assigned to Dimensions**
- **Fields with numeric values are assigned with Measures**

Dimensions and Measures in Tableau

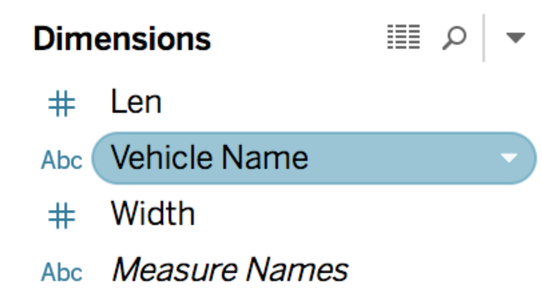
- **Depending on the contents, Tableau initially assigns each field in the data source to either:**
  - **Dimensions or**
  - **Measures**
- **Fields with categorical data such as names, dates, or geographical data are assigned to Dimensions**
- **Fields with numeric values are assigned with Measures**

- **Most fields can be used either as a Dimension or as a Measure and can be either continuous or discrete, according to the user requirements.**

Discrete vs. Continuous

- **When dragging fields from the Data pane into a view:**
  - **A field from the Dimensions area will usually be discrete**

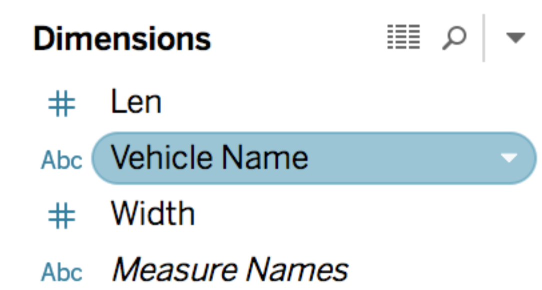
    (Blue background /)



Discrete vs. Continuous

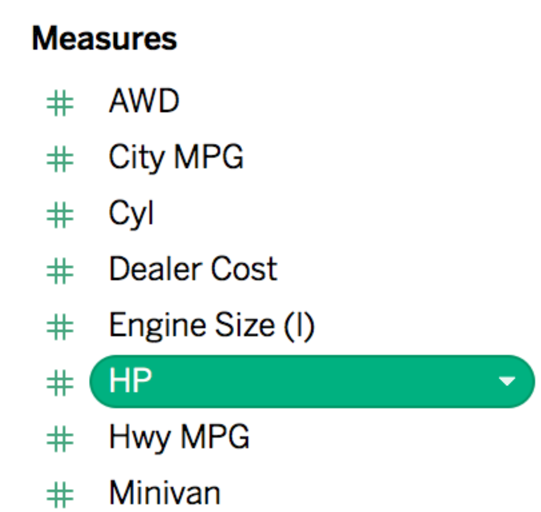
- **When dragging fields from the Data pane into a view:**
  - **A field from the Dimensions area will usually be discrete**

    (Blue background /)



  - **A field from the Measures area will usually be continuous**

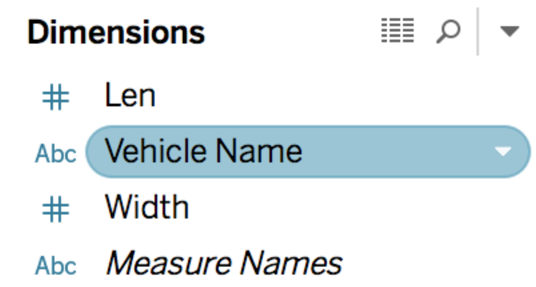
    (Green background /)



Discrete vs. Continuous

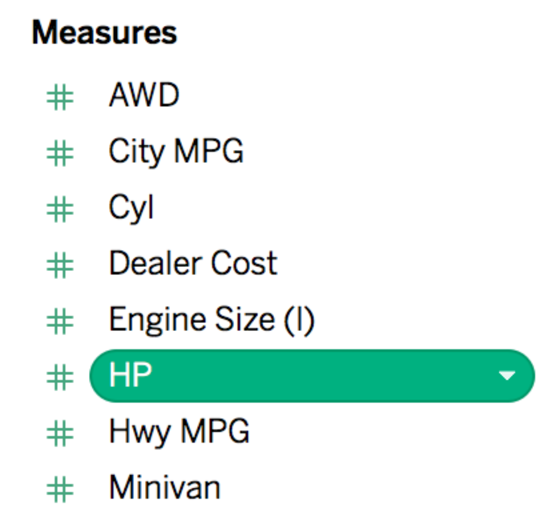
- **When dragging fields from the Data pane into a view:**
  - **A field from the Dimensions area will usually be discrete**

    (Blue background /)



  - **A field from the Measures area will usually be continuous**

    (Green background /)



- **This distinction discrete/continuous is important for:**
  - Rows/Columns - Headers vs. Axis
  - Filters - Individual vs. Interval
  - Visual Variable Mapping (Color, Size) - Distinct vs. Gradual

Dimensions and Measures vs. Discrete and Continuous

- **All combinations are possible:**

discrete dimensions	Product Name
continuous dimensions (possible only with Date dimensions)	⊞ QUARTER(Order Date)
discrete measures	SUM(Profit)
continuous measures	SUM(Profit)

Dimensions and Measures vs. Discrete and Continuous

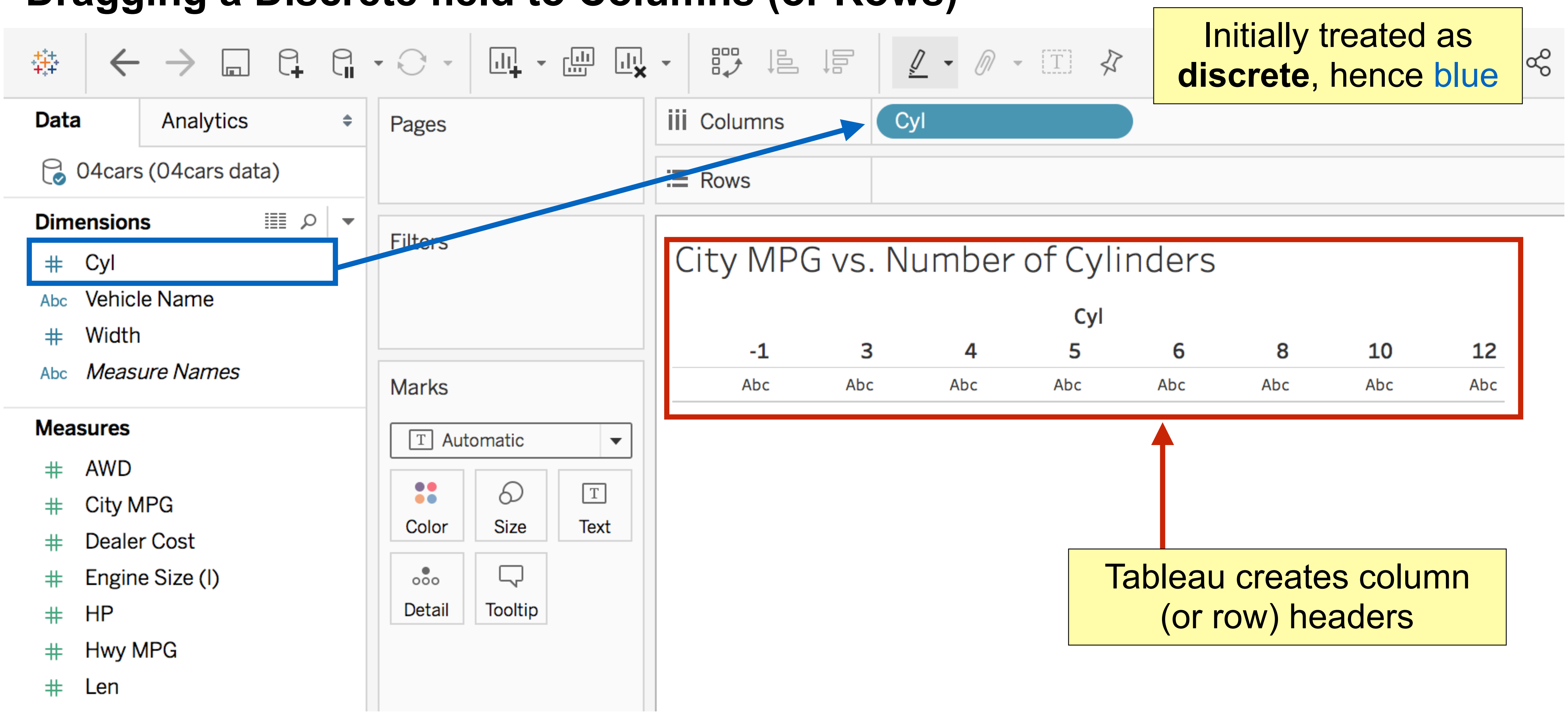
- **All combinations are possible:**

discrete dimensions	Product Name
continuous dimensions (possible only with Date dimensions)	⊞ QUARTER(Order Date)
discrete measures	SUM(Profit)
continuous measures	SUM(Profit)

- **But:**
  - **Measures (discrete or continuum) will still aggregate data**
  - **Dimensions (discrete or continuum) will not aggregate data**

Working with Discrete (in Rows/Columns)

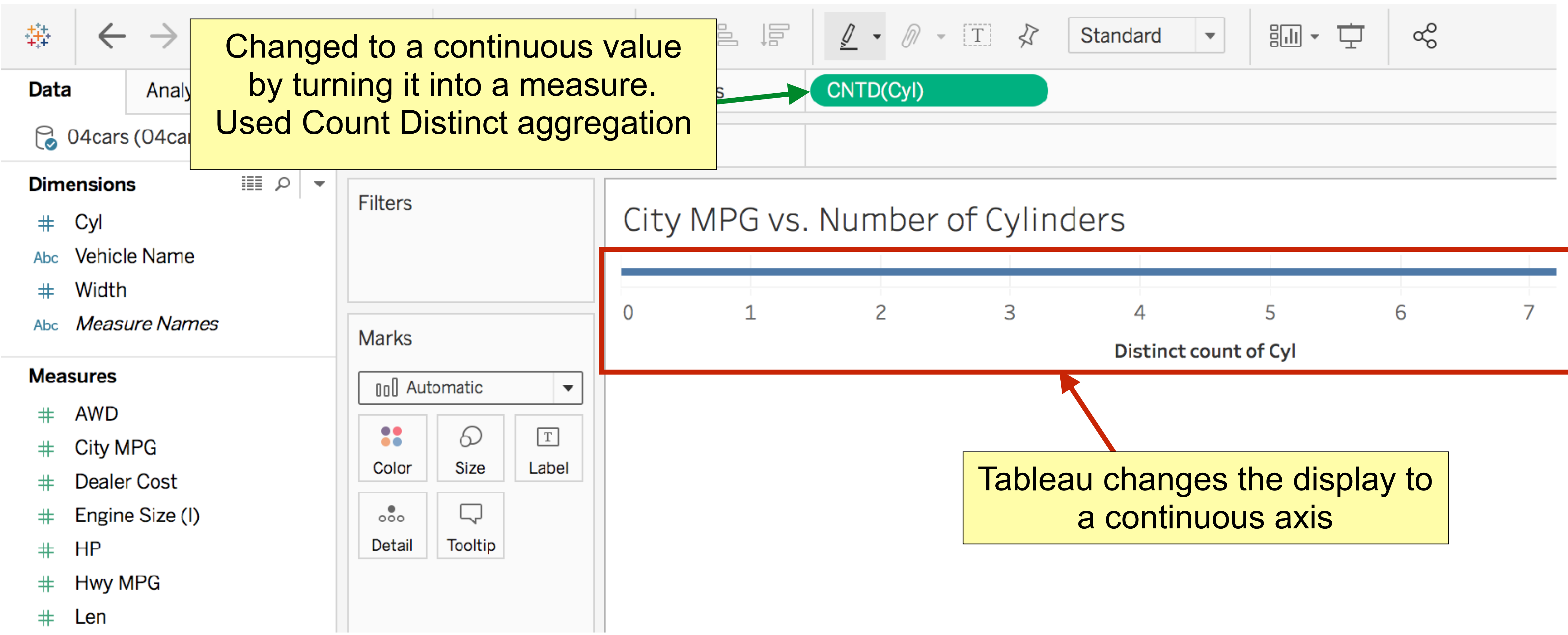
- **Dragging a Discrete field to Columns (or Rows)**



- **Date or Numeric Dimension fields can be made continuous. Other Dimension fields can become continuous by using some aggregation functions: Count…**

Working with Discrete (in Rows/Columns)

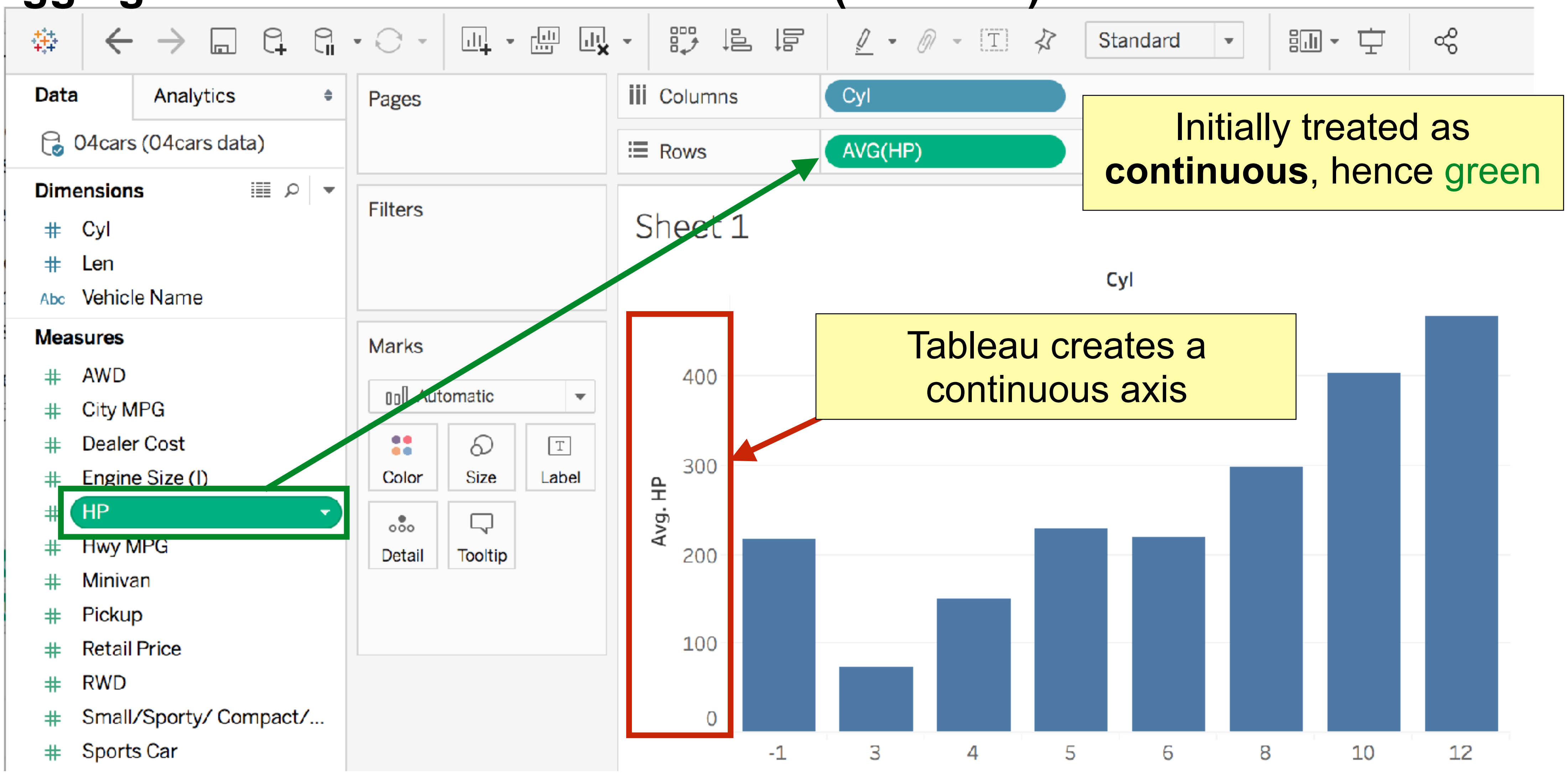
- **Changing a field already in Columns (or Rows) into a measure**



- **In Tableau queries, dimensions in the view are expressed in SQL as "`Group By`" clauses.**

Working with Continuous (in Rows/Columns)

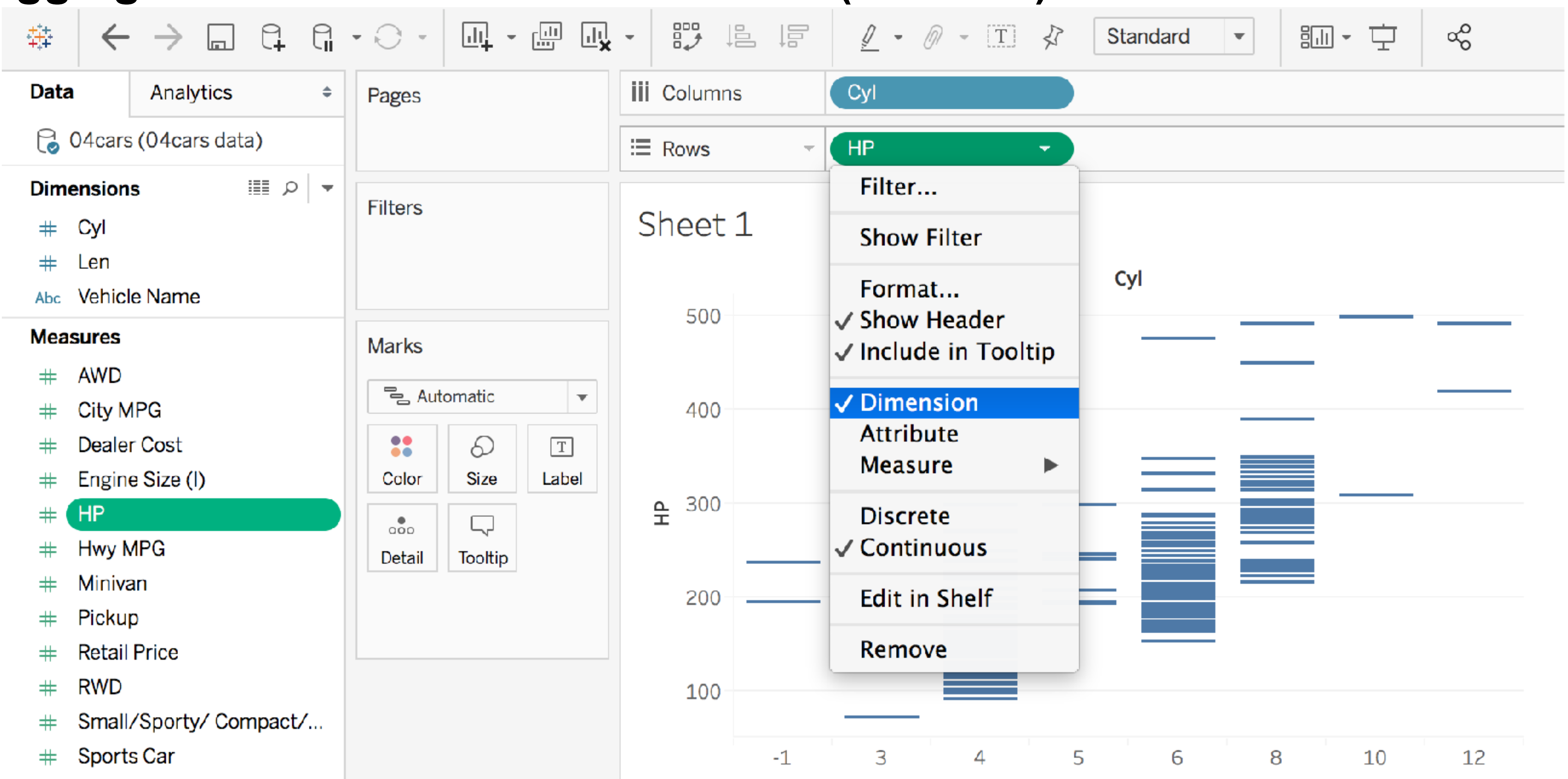
- **Dragging a Continuous field to Columns (or Rows)**



- **Later changing it to Discrete will turn the axis into column (or rows) headers.**
- **Tableau still aggregates values. Measures are normally aggregated.**

Working with Continuous (in Rows/Columns)

- **Dragging a Continuous field to Columns (or Rows)**

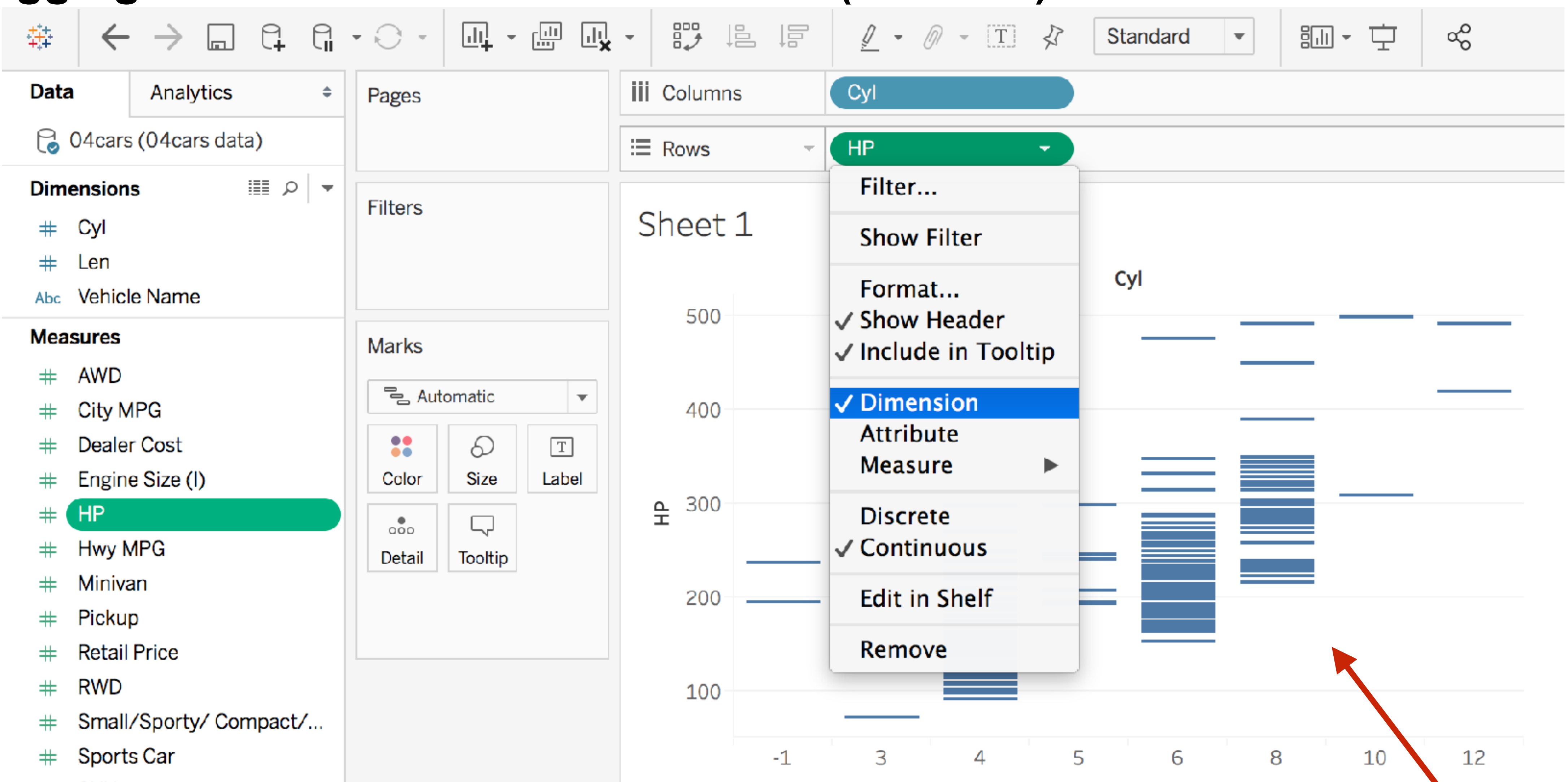


- **Turning the field in the view into a Dimension will aggregate by value!**

  Note that `AVG(HP)` became `HP`.

Working with Continuous (in Rows/Columns)

- **Dragging a Continuous field to Columns (or Rows)**

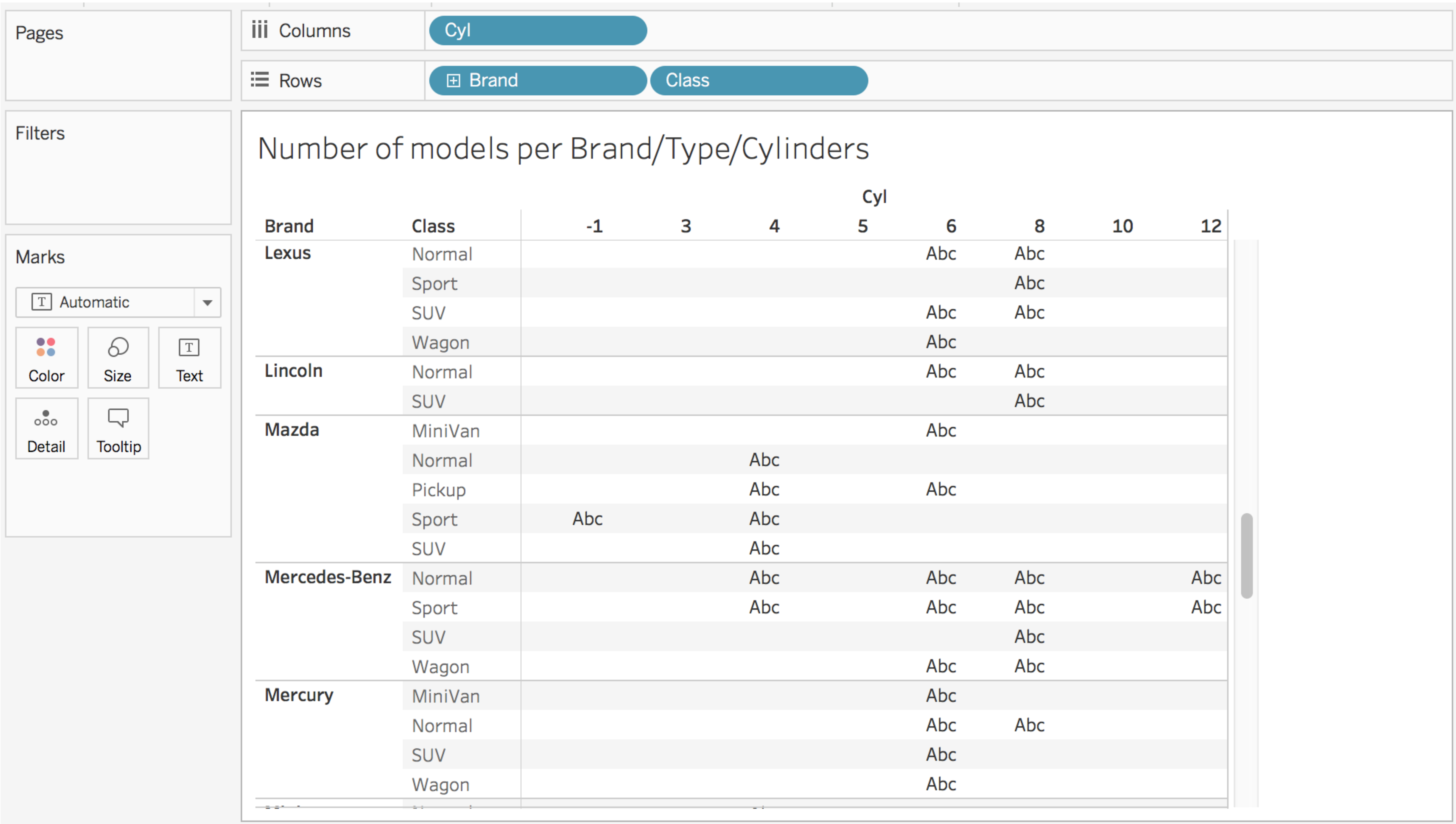


- **Turning the field in the view into a Dimension will aggregate by value!**

Note that `AVG(HP)` became `HP`.

What happened?What does a mark represent?

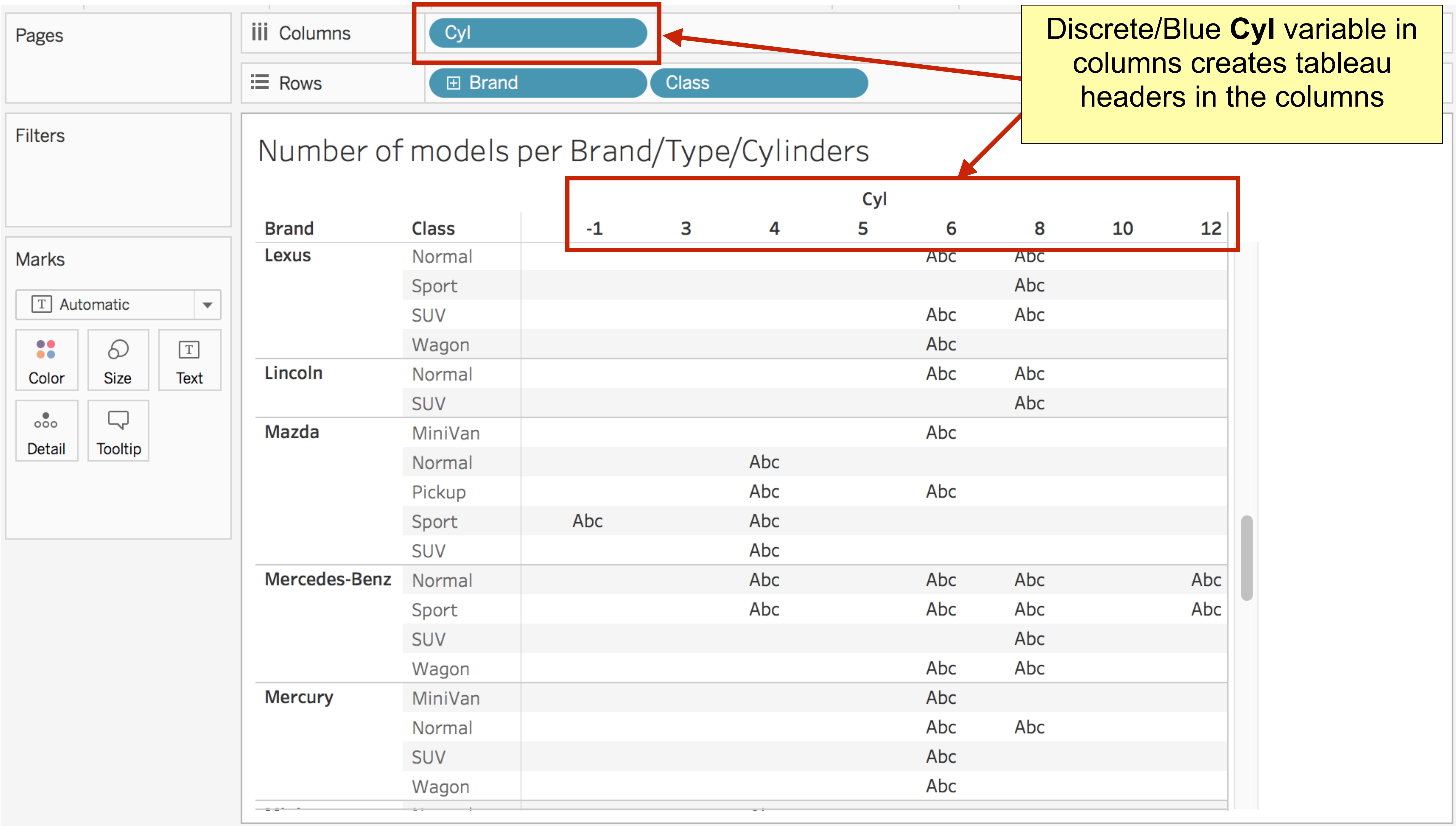
Examples: discrete



Number of models per Brand/Type/Cylinders

		Cyl							
Brand	**Class**	**-1**	**3**	**4**	**5**	**6**	**8**	**10**	**12**
Lexus	Normal					Abc	Abc		
	Sport						Abc		
	SUV					Abc	Abc		
	Wagon					Abc			
Lincoln	Normal					Abc	Abc		
	SUV						Abc		
Mazda	MiniVan					Abc			
	Normal			Abc					
	Pickup			Abc		Abc			
	Sport	Abc		Abc					
	SUV			Abc					
Mercedes-Benz	Normal			Abc		Abc	Abc		Abc
	Sport			Abc		Abc	Abc		Abc
	SUV						Abc		
	Wagon					Abc	Abc		
Mercury	MiniVan					Abc			
	Normal					Abc	Abc		
	SUV					Abc			
	Wagon					Abc			

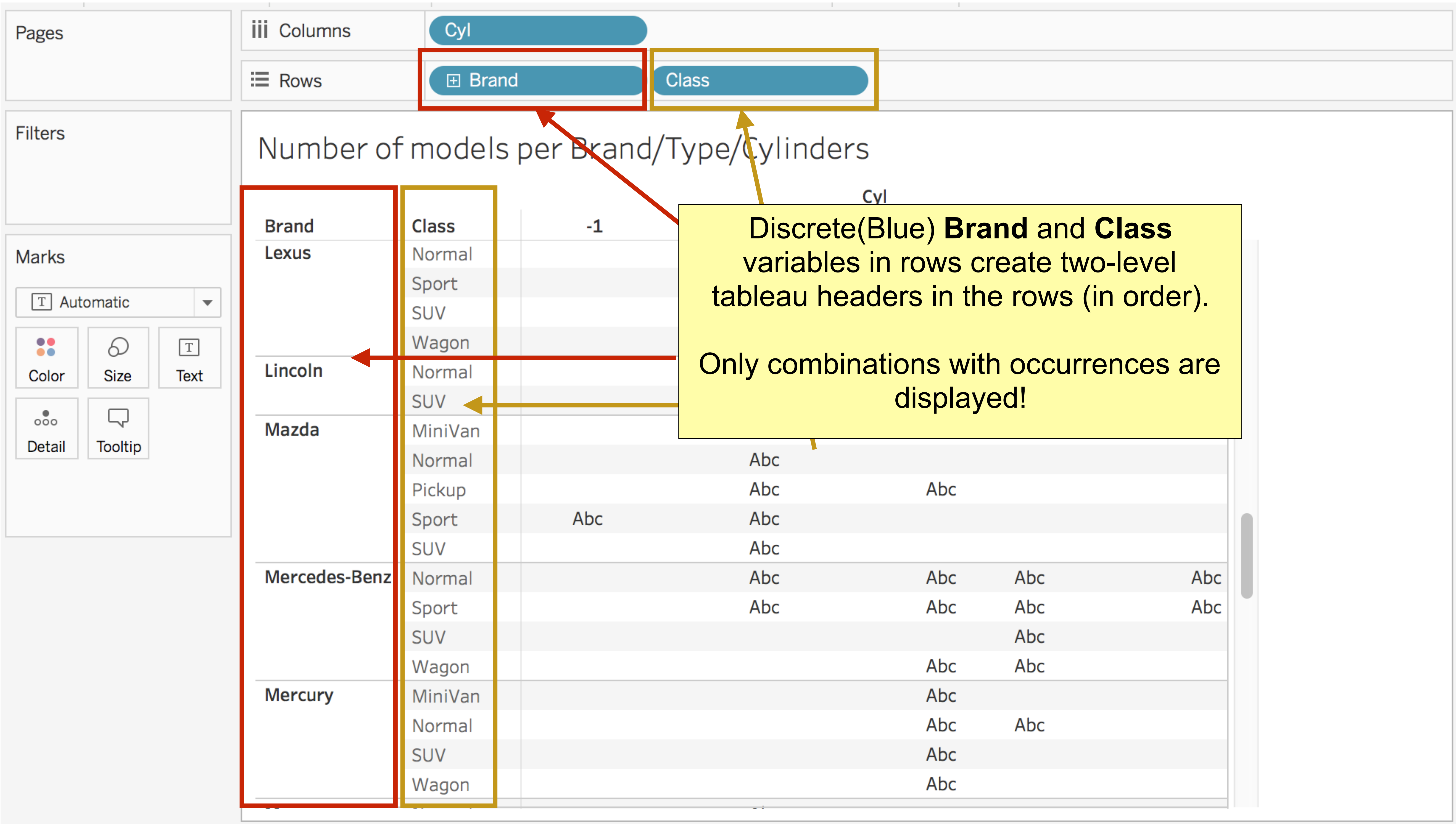
Examples: discrete



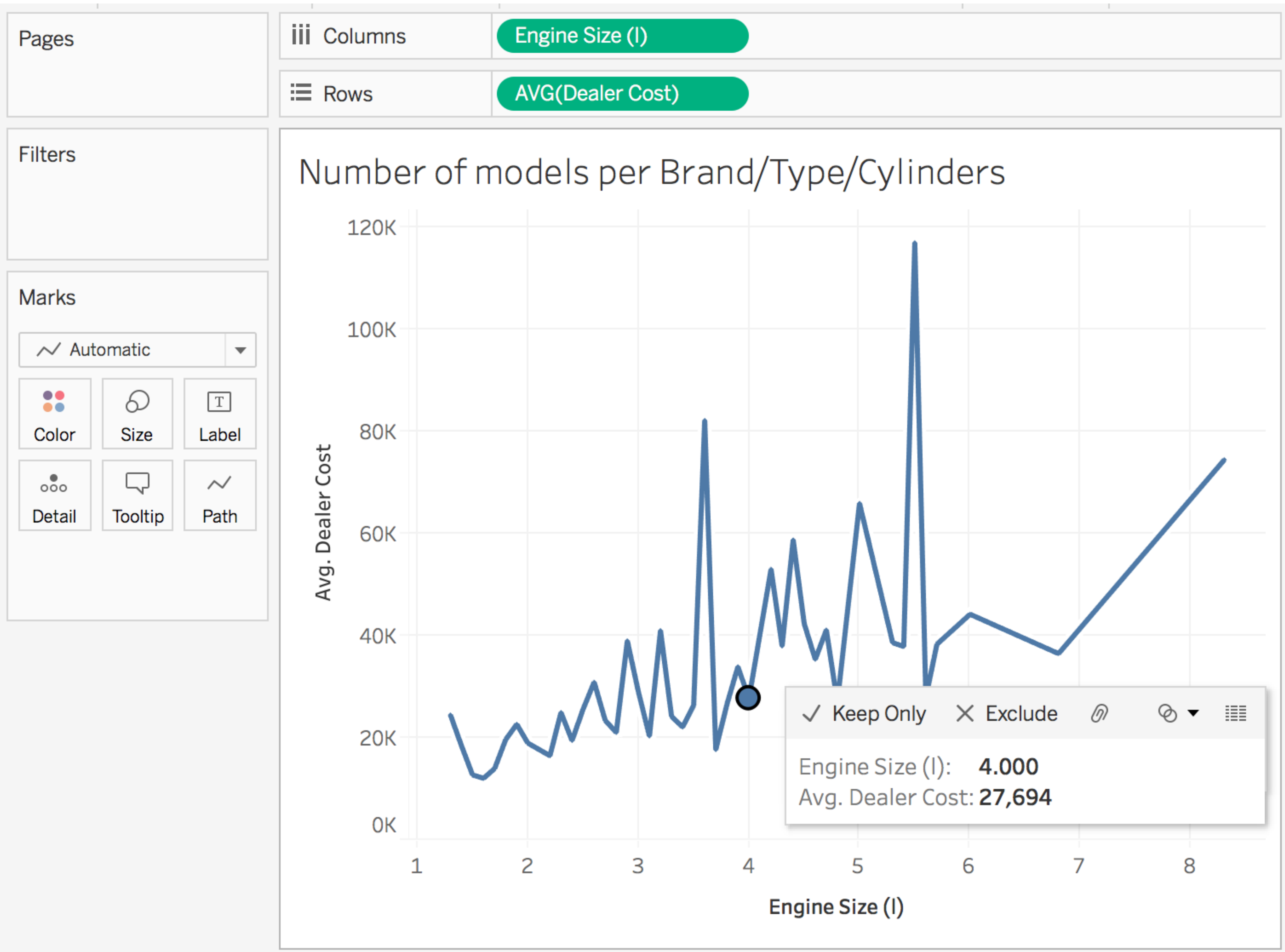
Brand	Class	-1	3	4	5	6	8	10	12
Lexus	Normal					Abc	Abc		
	Sport						Abc		
	SUV					Abc	Abc		
	Wagon					Abc			
Lincoln	Normal					Abc	Abc		
	SUV						Abc		
Mazda	MiniVan					Abc			
	Normal			Abc					
	Pickup			Abc		Abc			
	Sport	Abc		Abc					
	SUV			Abc					
Mercedes-Benz	Normal			Abc		Abc	Abc		Abc
	Sport			Abc		Abc	Abc		Abc
	SUV						Abc		
	Wagon					Abc	Abc		
Mercury	MiniVan					Abc			
	Normal					Abc	Abc		
	SUV					Abc			
	Wagon					Abc			

Discrete/Blue **Cyl** variable in columns creates tableau headers in the columns

Examples: discrete

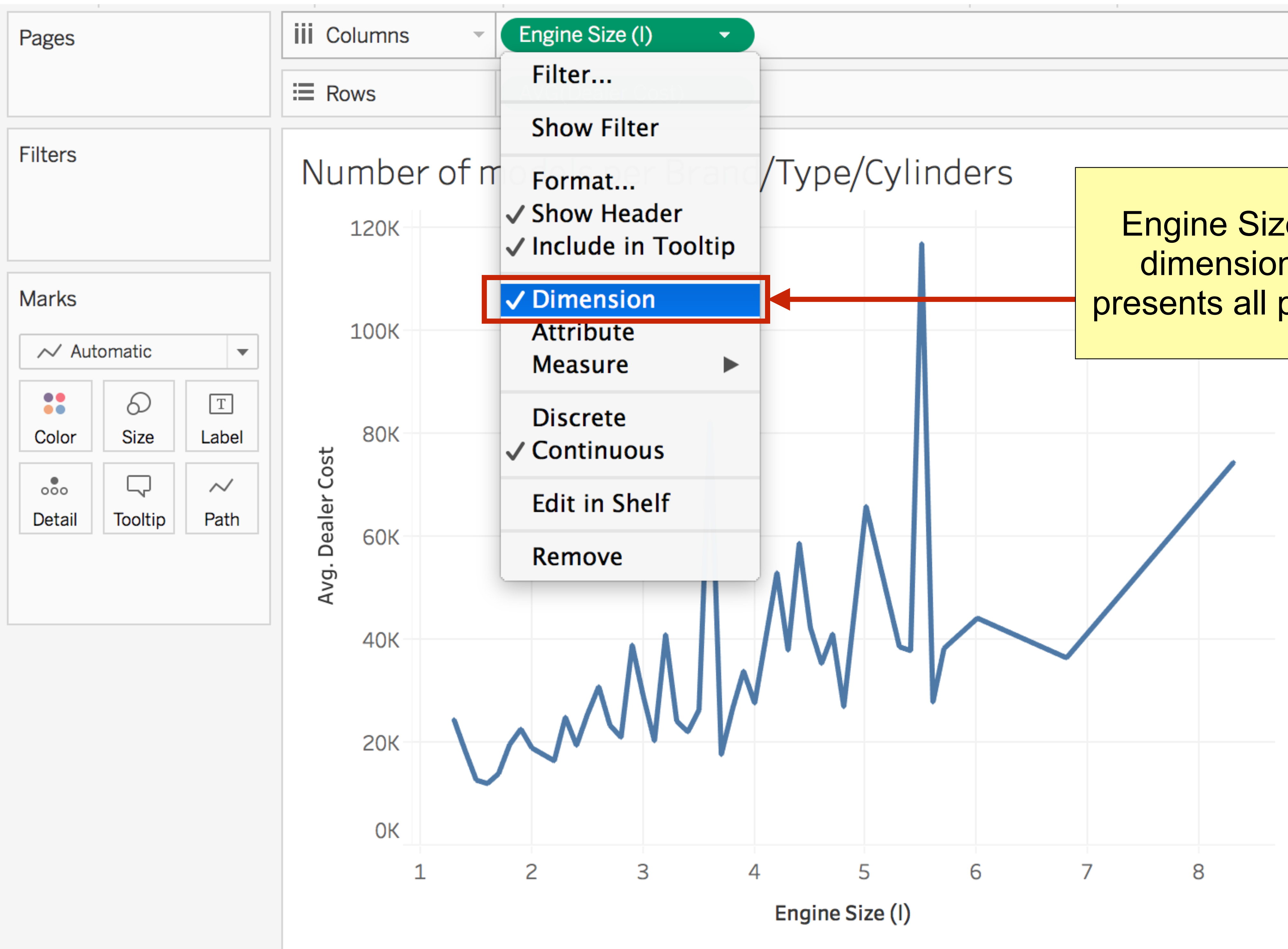


Examples: continuous



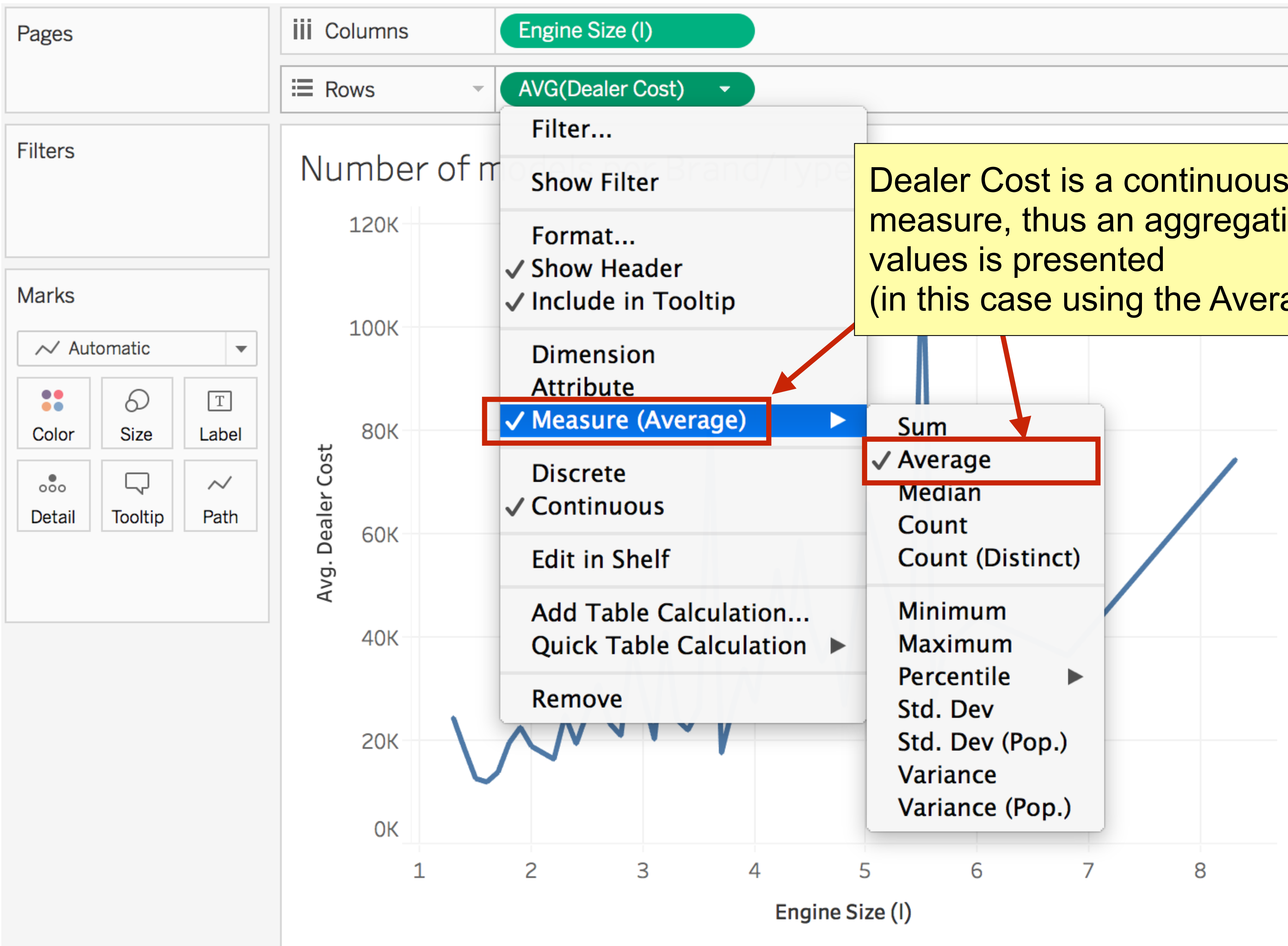
Examples: continuous

Examples: continuous - dimension



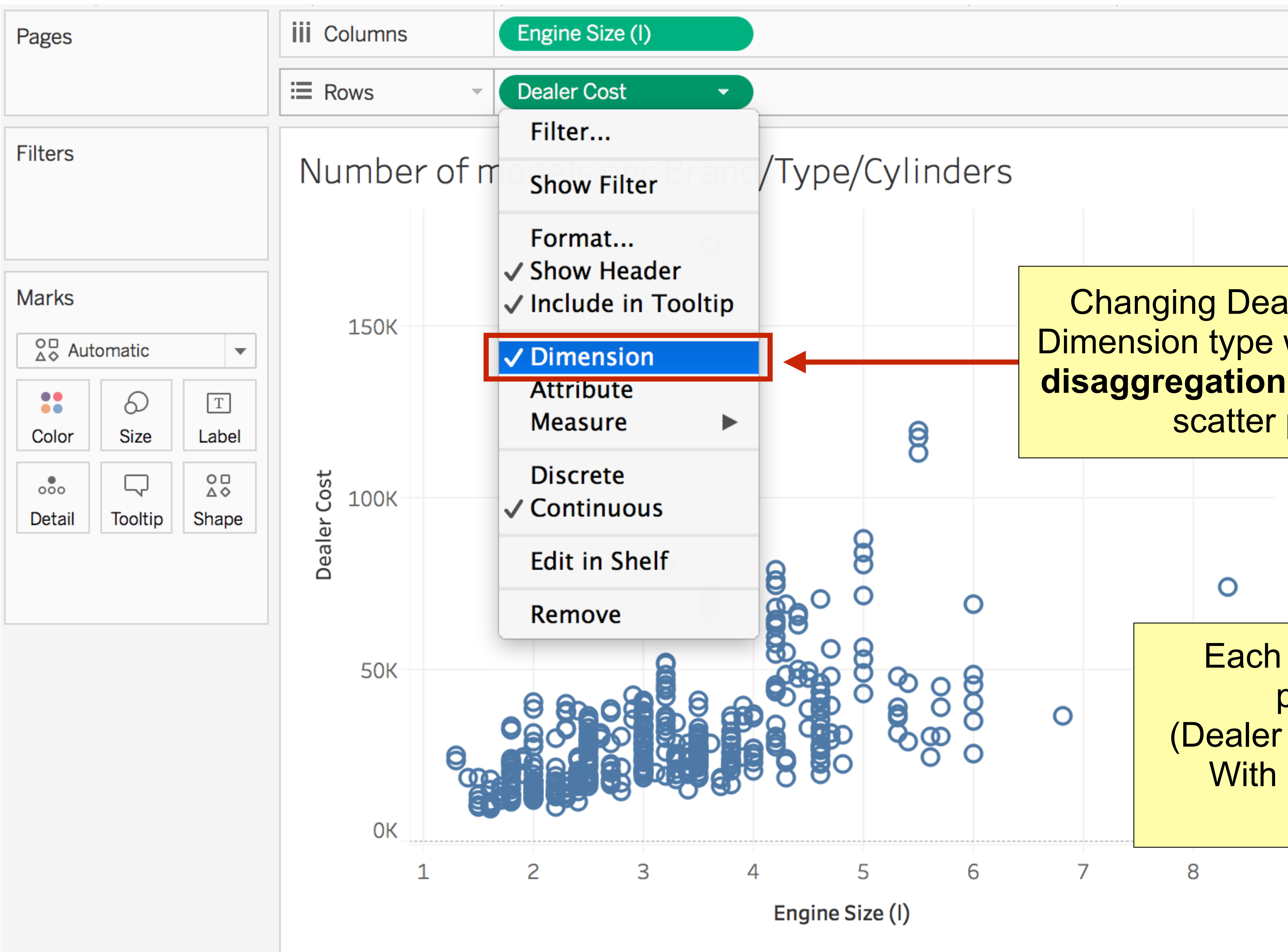
Engine Size (l) here is a dimension so tableau presents all possible values

Examples: continuous - measure



Dealer Cost is a continuous measure, thus an aggregation of values is presented
(in this case using the Average)

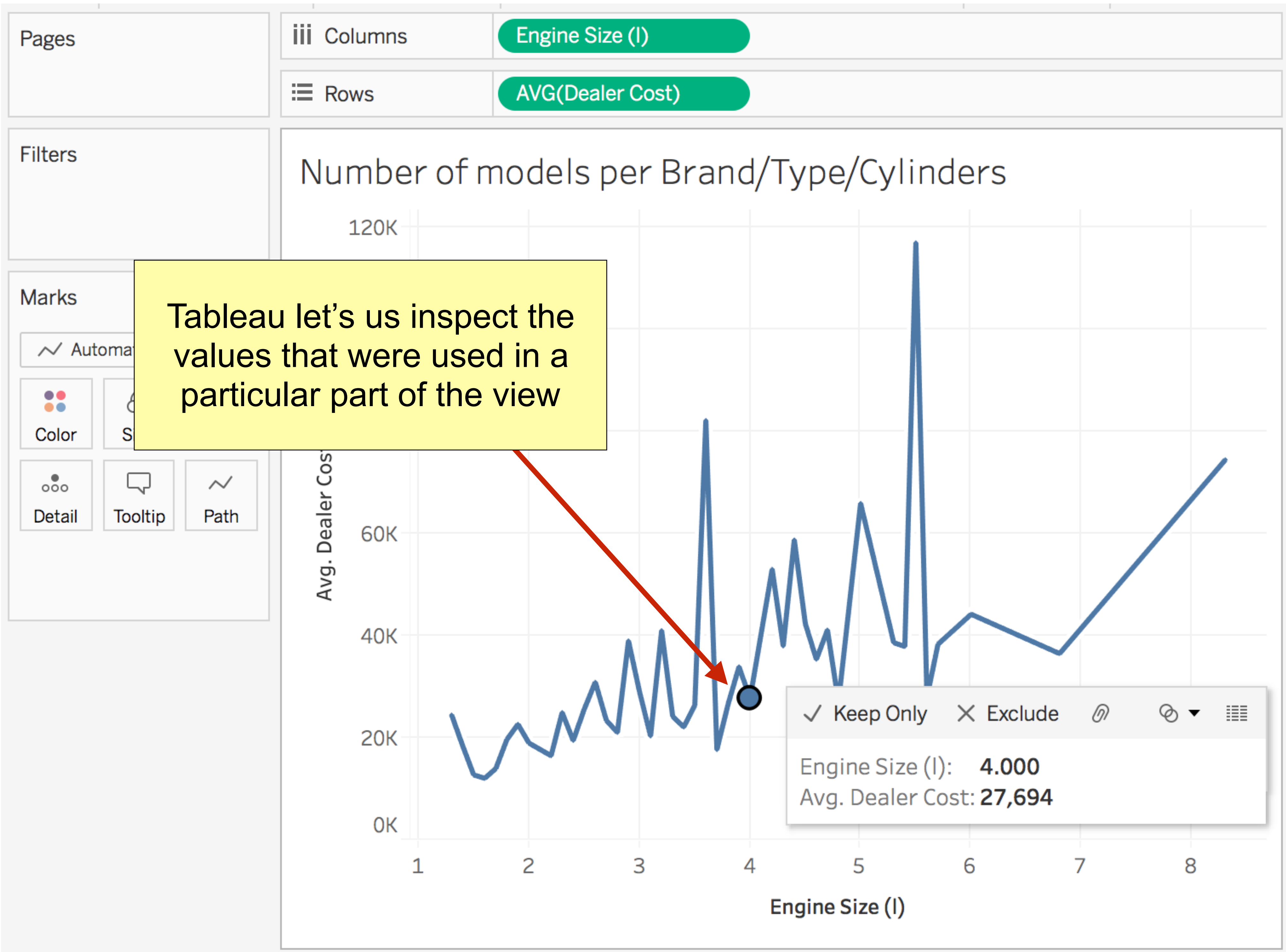
Examples: continuous - dimensions



Changing Dealer Cost to Dimension type will do **some disaggregation** producing a scatter plot

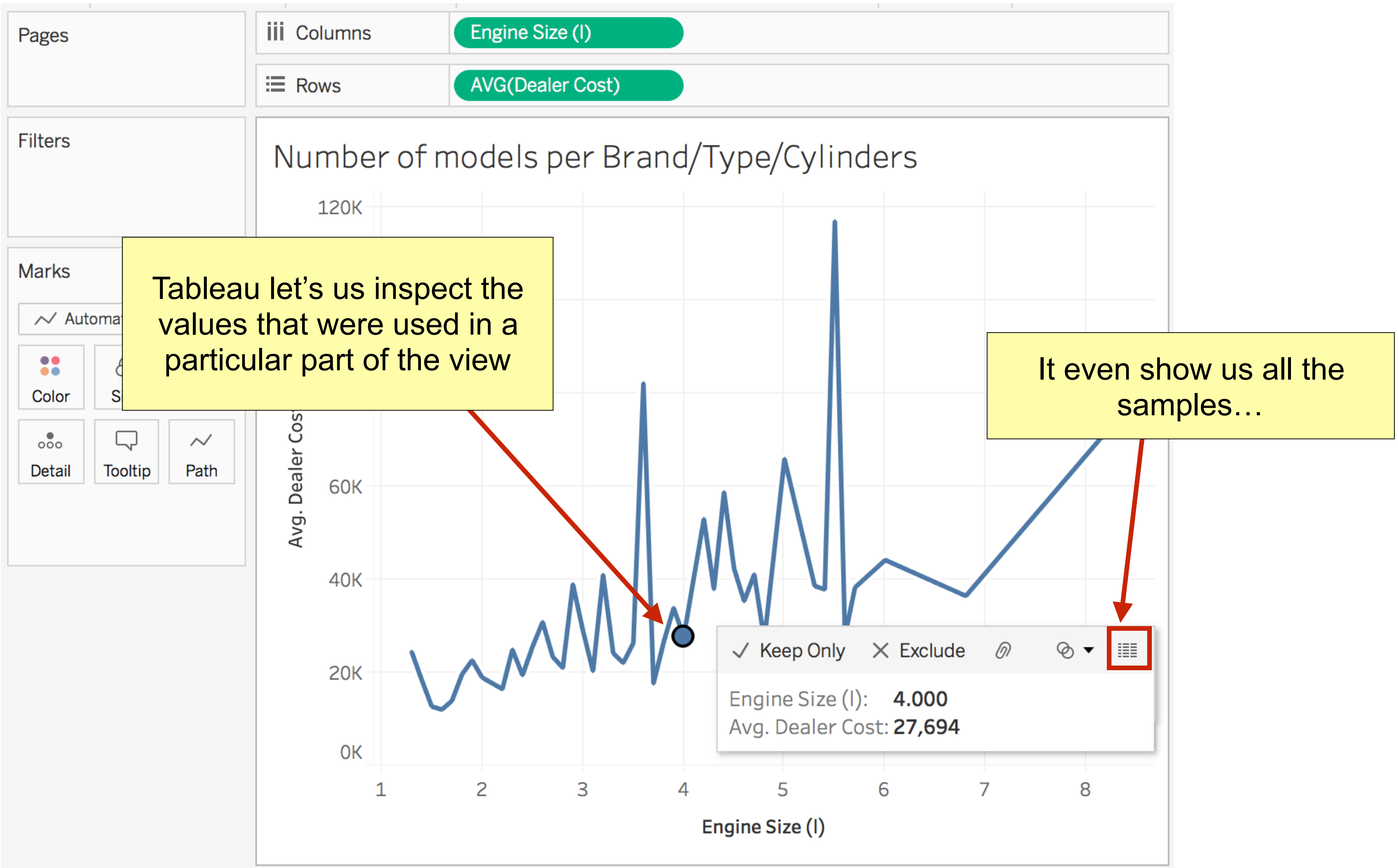
Each dot represents a possible pair (Dealer Cost, Engine Size) With possibly >1 data samples!

Data Inspection

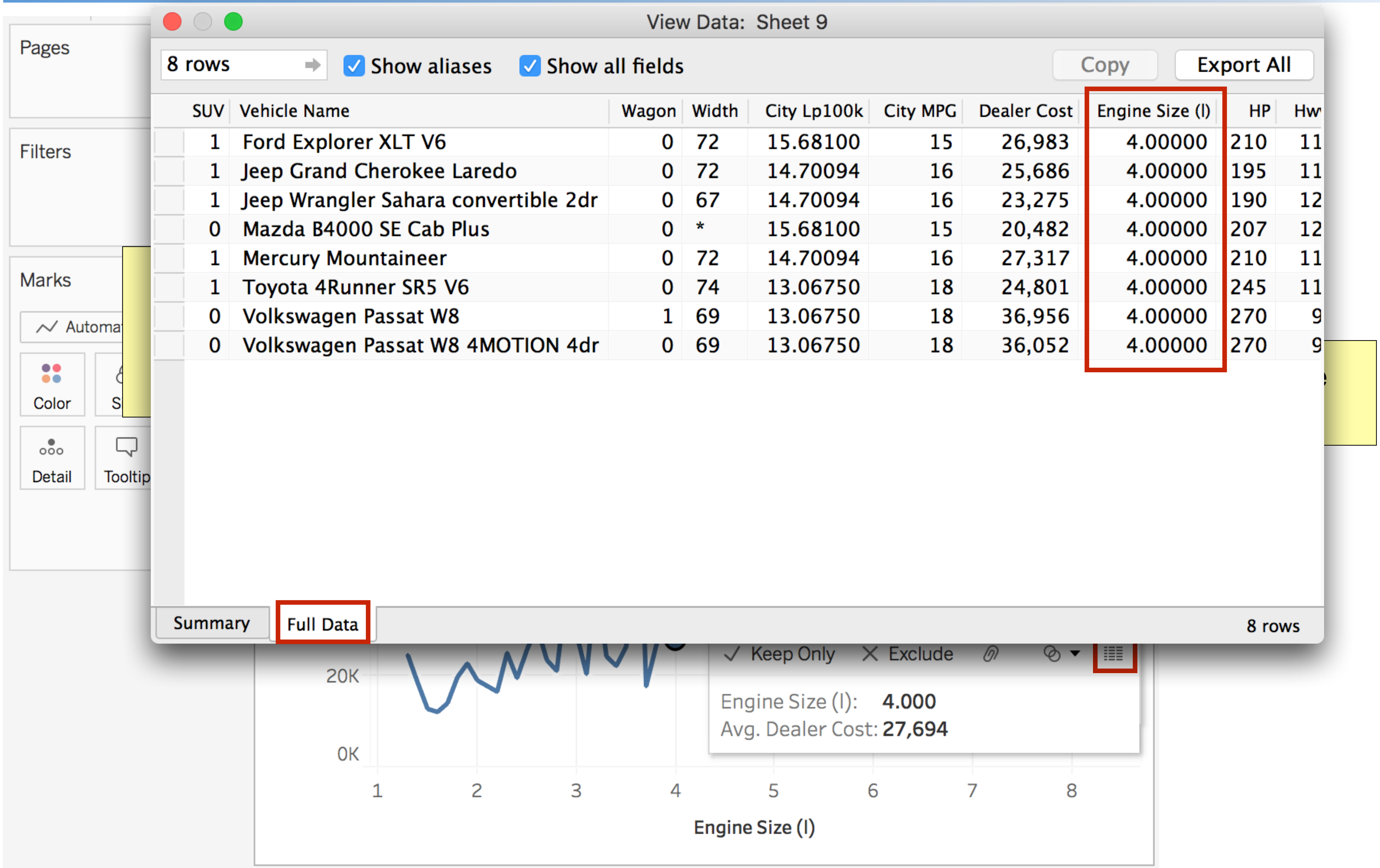


FACULTADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

Data Inspection



Data Inspection

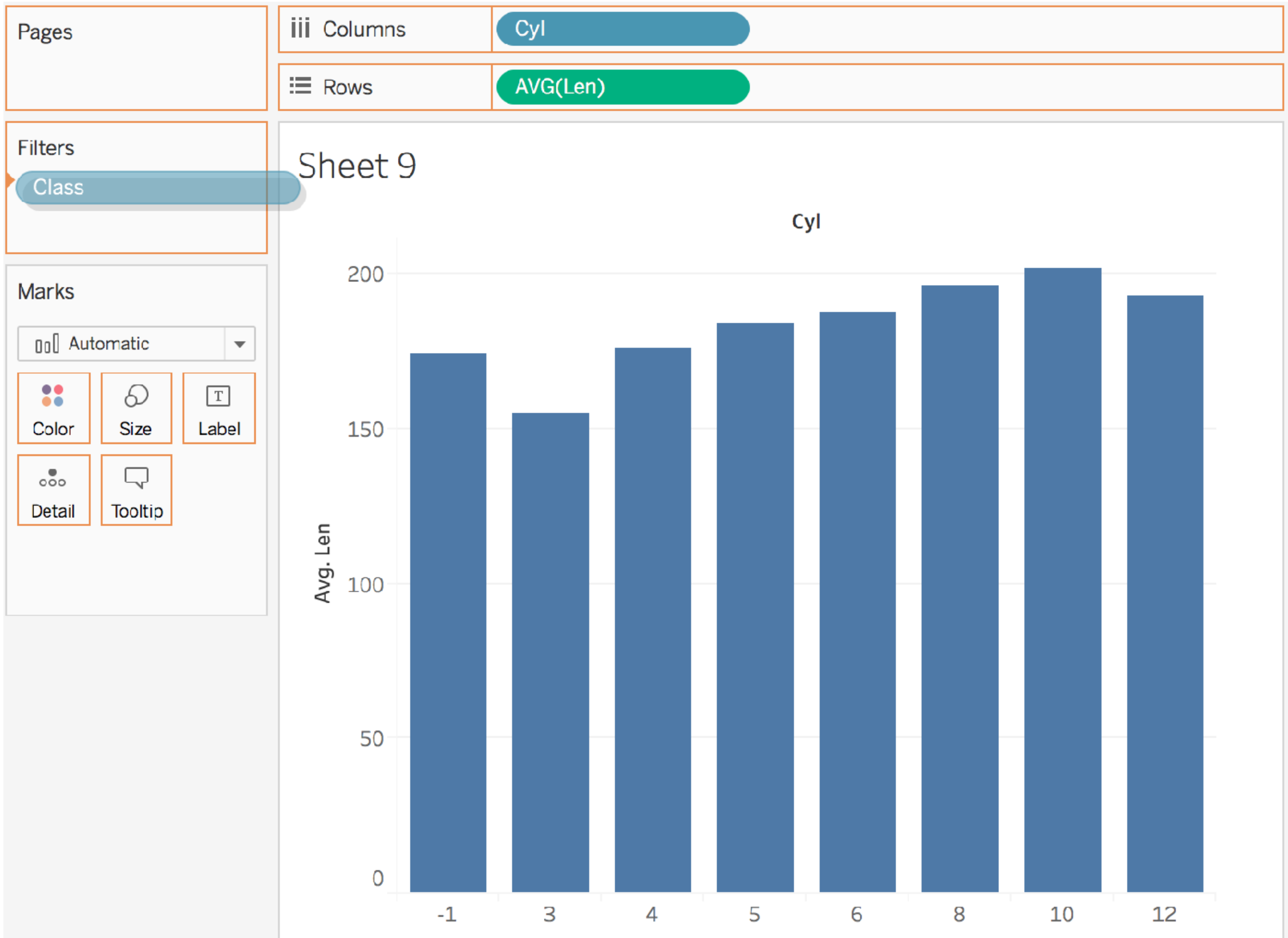


SUV	Vehicle Name	Wagon	Width	City Lp100k	City MPG	Dealer Cost	Engine Size (l)	HP	Hw
1	Ford Explorer XLT V6	0	72	15.68100	15	26,983	4.00000	210	11
1	Jeep Grand Cherokee Laredo	0	72	14.70094	16	25,686	4.00000	195	11
1	Jeep Wrangler Sahara convertible 2dr	0	67	14.70094	16	23,275	4.00000	190	12
0	Mazda B4000 SE Cab Plus	0	*	15.68100	15	20,482	4.00000	207	12
1	Mercury Mountaineer	0	72	14.70094	16	27,317	4.00000	210	11
1	Toyota 4Runner SR5 V6	0	74	13.06750	18	24,801	4.00000	245	11
0	Volkswagen Passat W8	1	69	13.06750	18	36,956	4.00000	270	9
0	Volkswagen Passat W8 4MOTION 4dr	0	69	13.06750	18	36,052	4.00000	270	9

FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

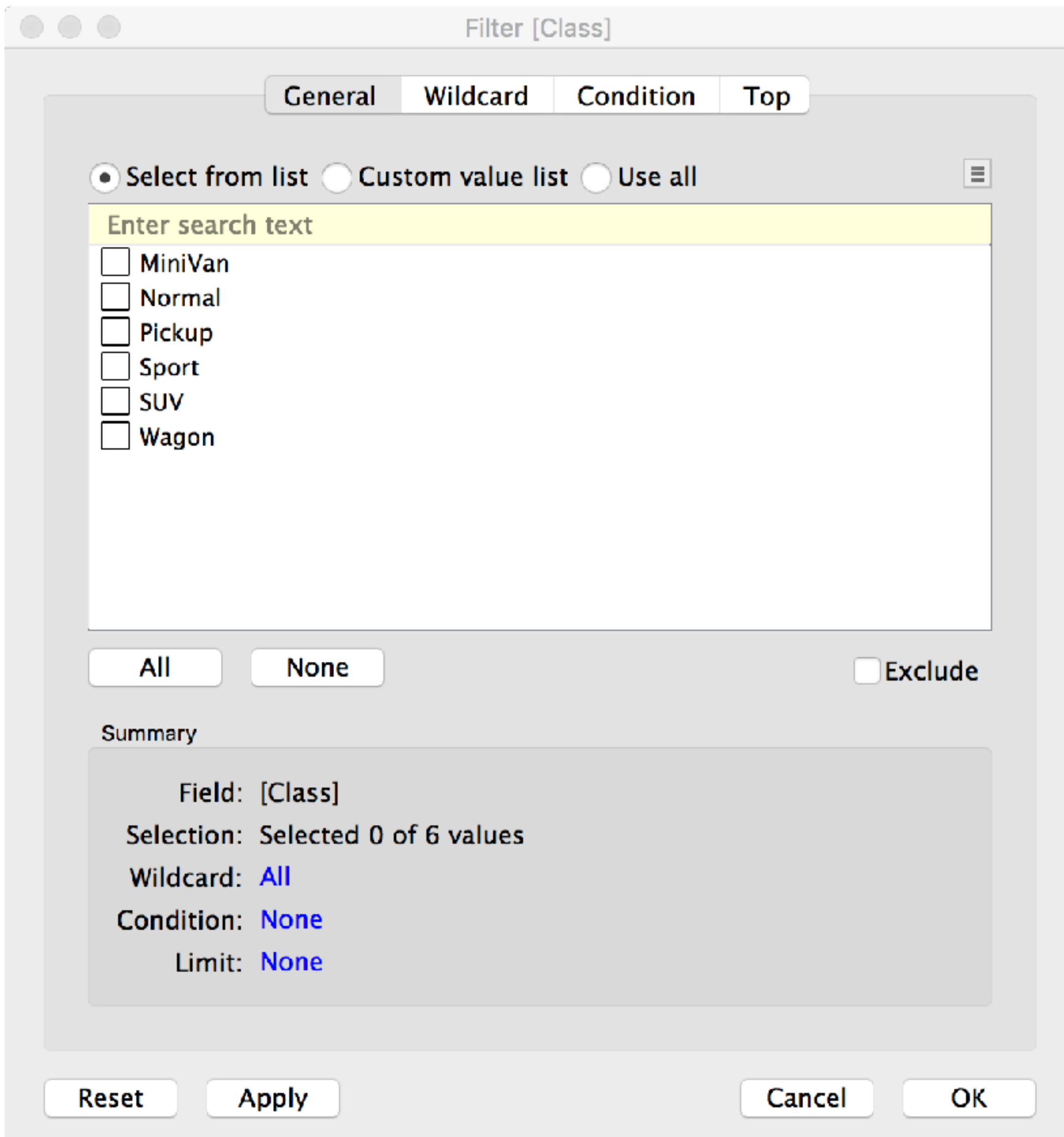
Filters: working with Discrete

- **Dragging a Discrete field to the Filters Card…**



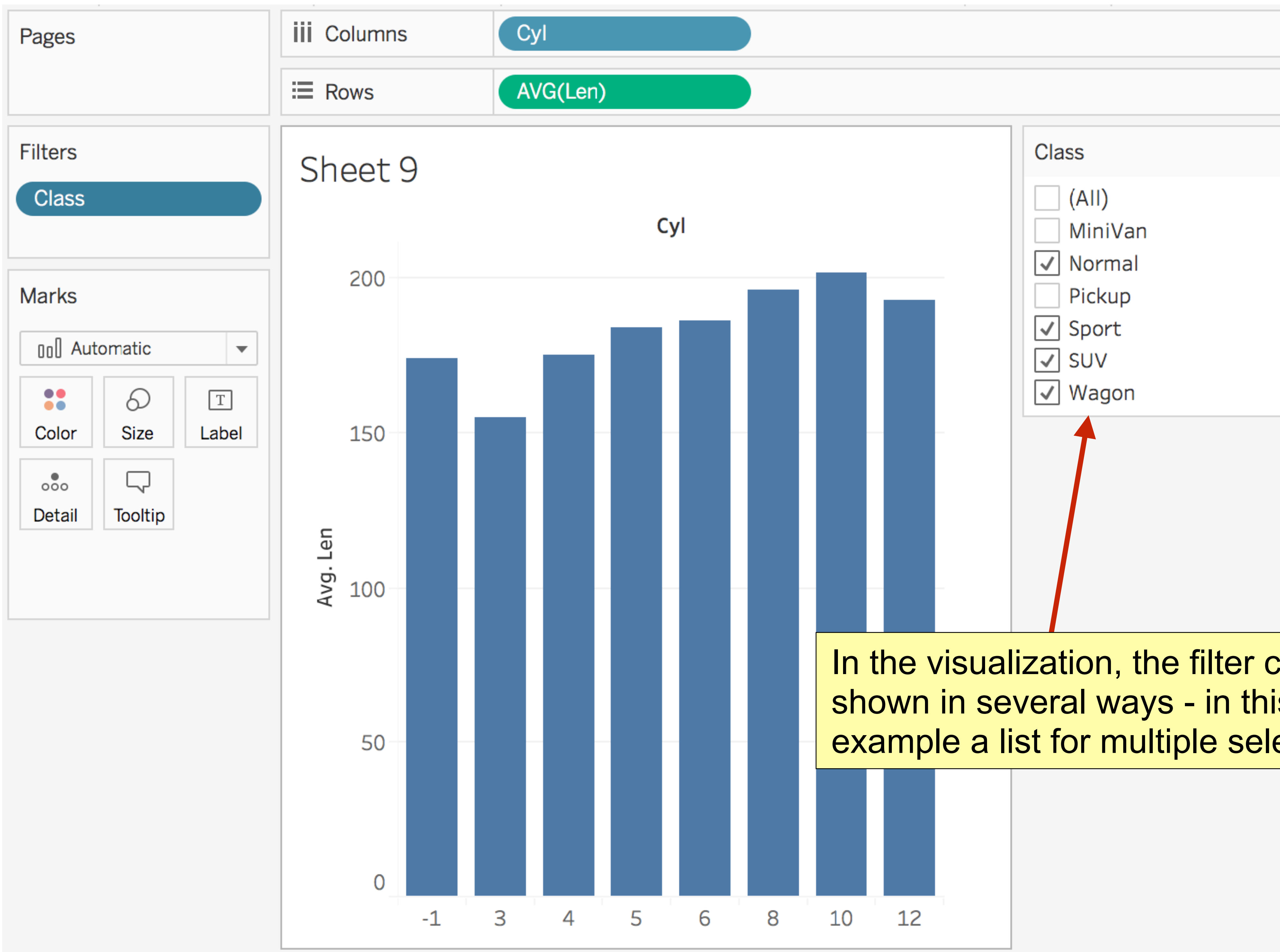
Filters: working with Discrete

- **Dragging a Discrete field to the Filters Card…**
- **Tableau will let you choose which values you want to use**



Since the data field is categorical we can choose the values by selecting them individually. There is no known order relation between them.

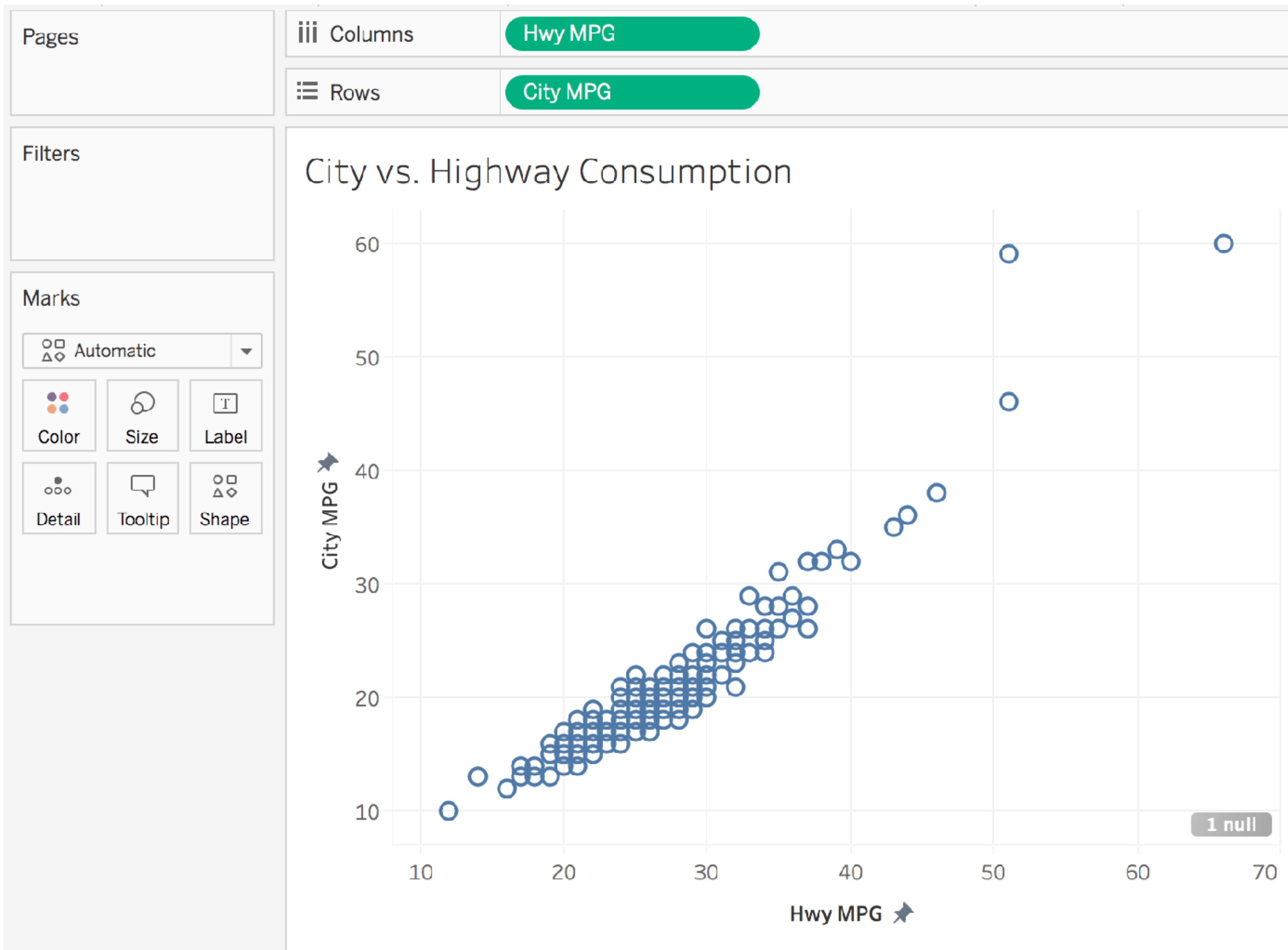
Filters: working with Discrete



In the visualization, the filter can be shown in several ways - in this example a list for multiple selection.

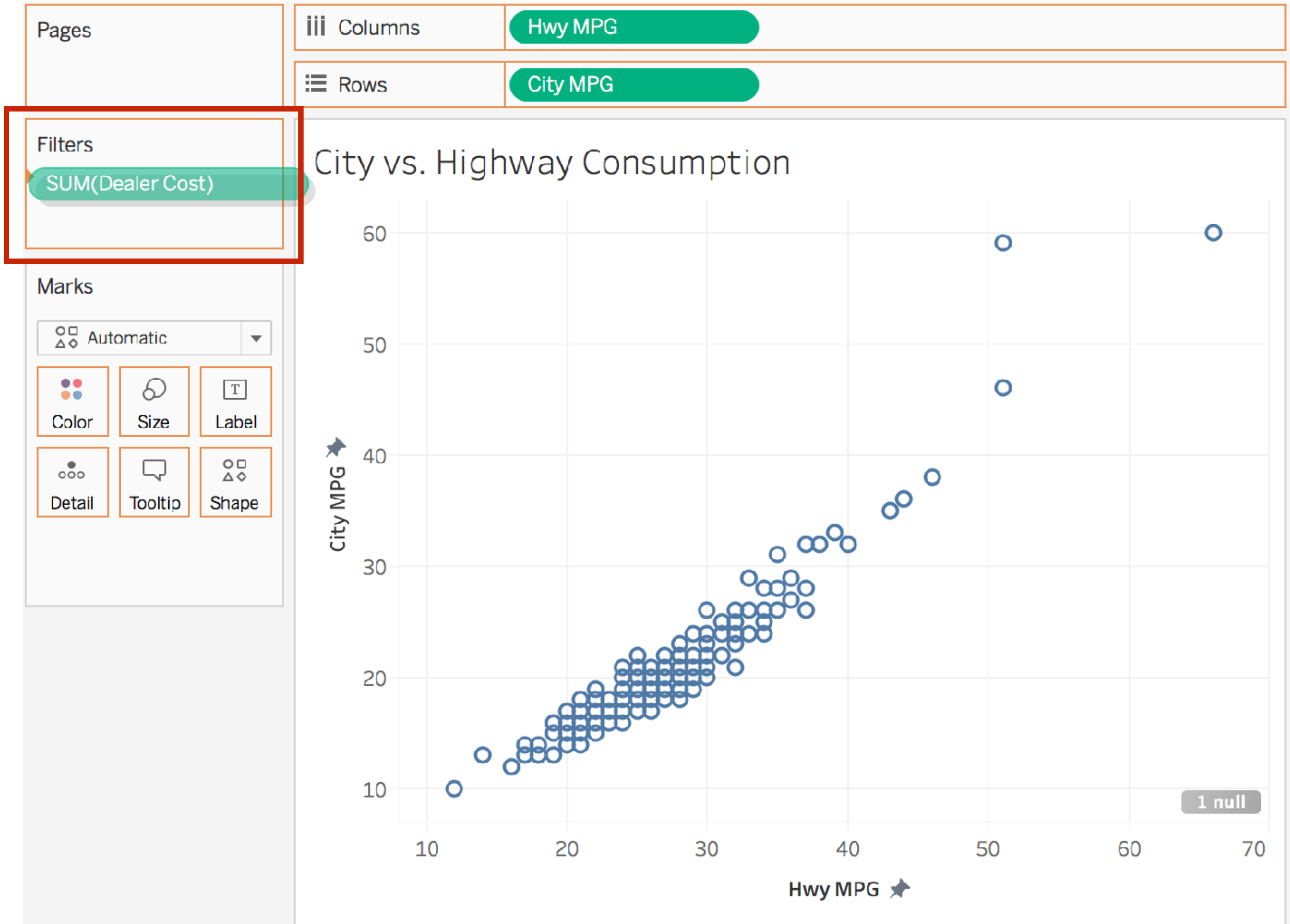
Filters: working with Continuous

- **Let's start with a scatter plot of City vs. Highway consumption:**



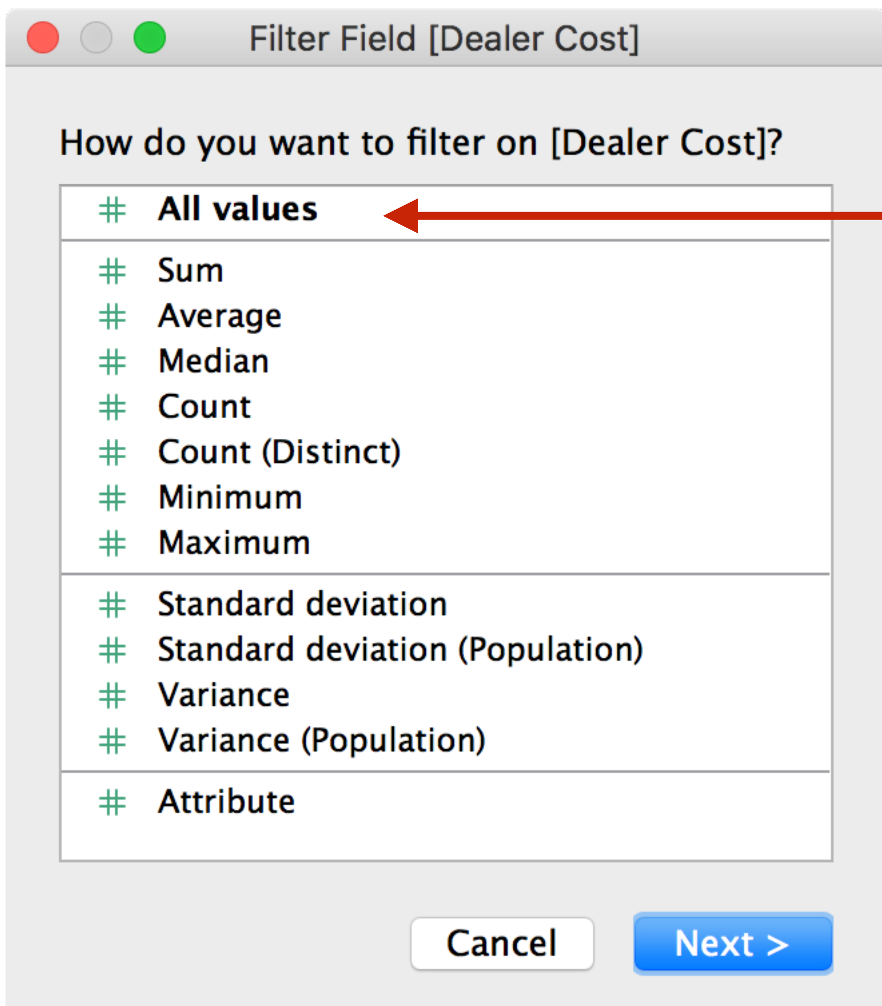
Filters: working with Continuous

- **Dragging a Continuous field to the Filters Card…**



Filters: working with Continuous

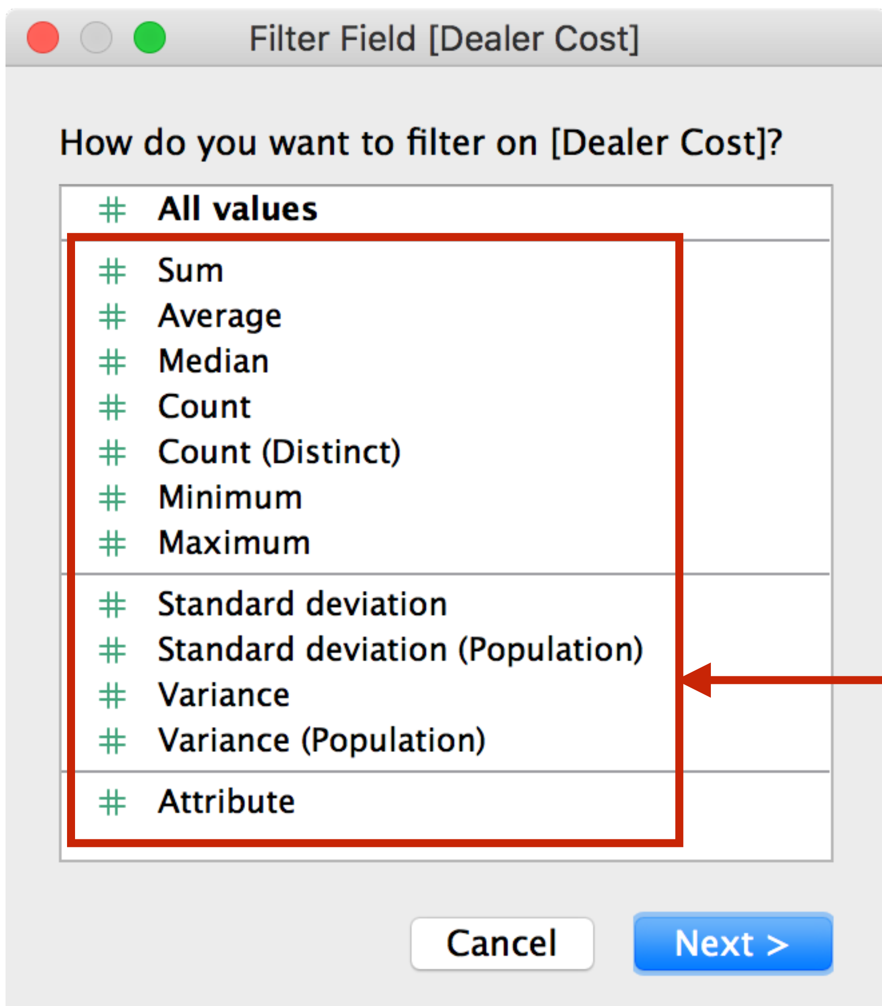
- **Dragging a Continuous field to the Filters Card**
- **Tableau will let you choose how to filter your data…**



Choose **All Values** if you want to filter on the raw data. This causes Tableau to compare your filter settings with the value which is held in each row of your data (for "Dealer Cost" in this case).

Filters: working with Continuous

- **Dragging a Continuous field to the Filters Card**
- **Tableau will let you choose how to filter your data…**

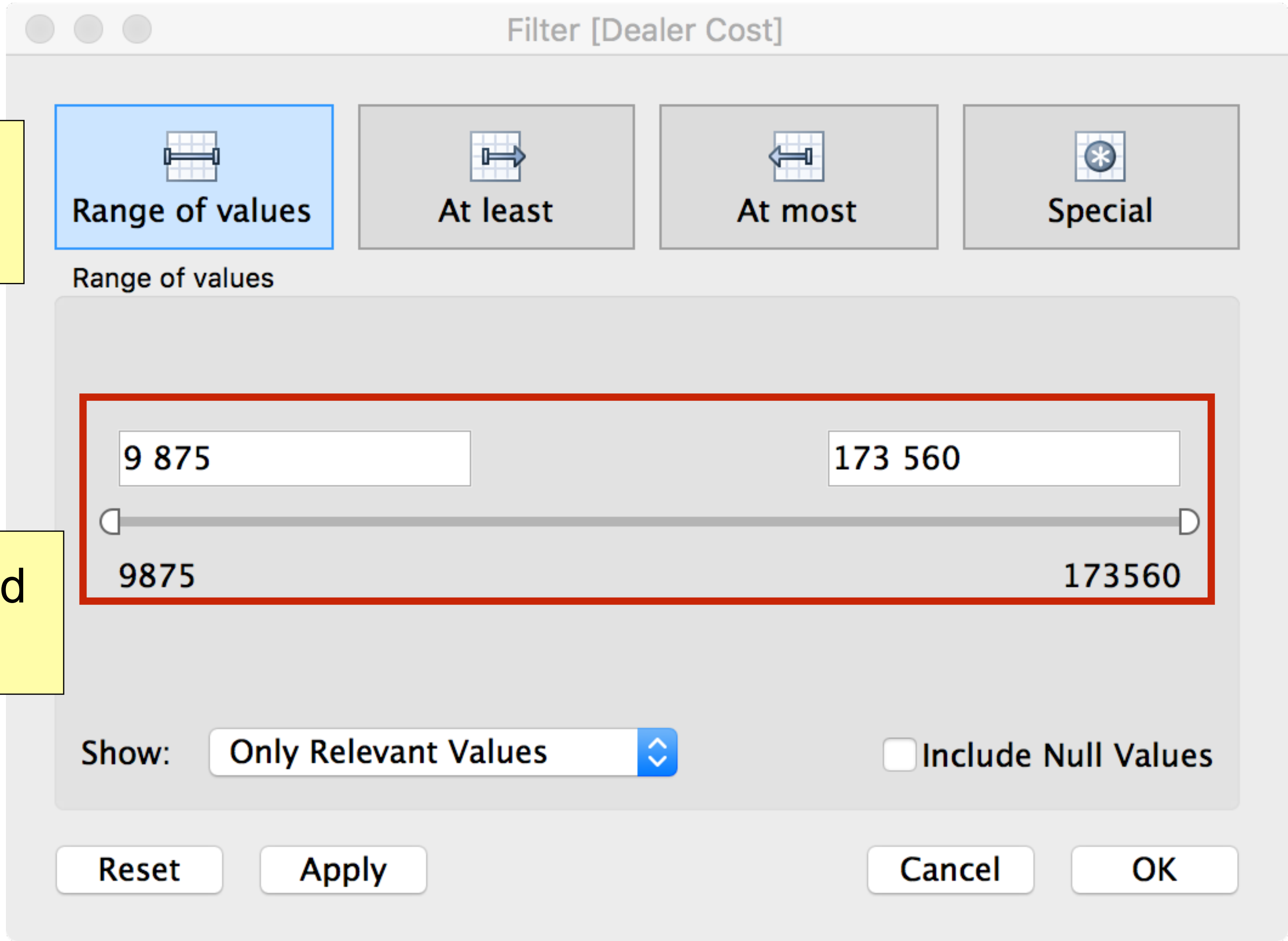


Choose an aggregation if you want to filter out 'members' of the dimensions by which you aggregate the data. For example, if you choose to aggregate by **Brand** and you are filtering on MIN('**Dealer Cost**') then your filter settings may exclude brands which have cheap cars…

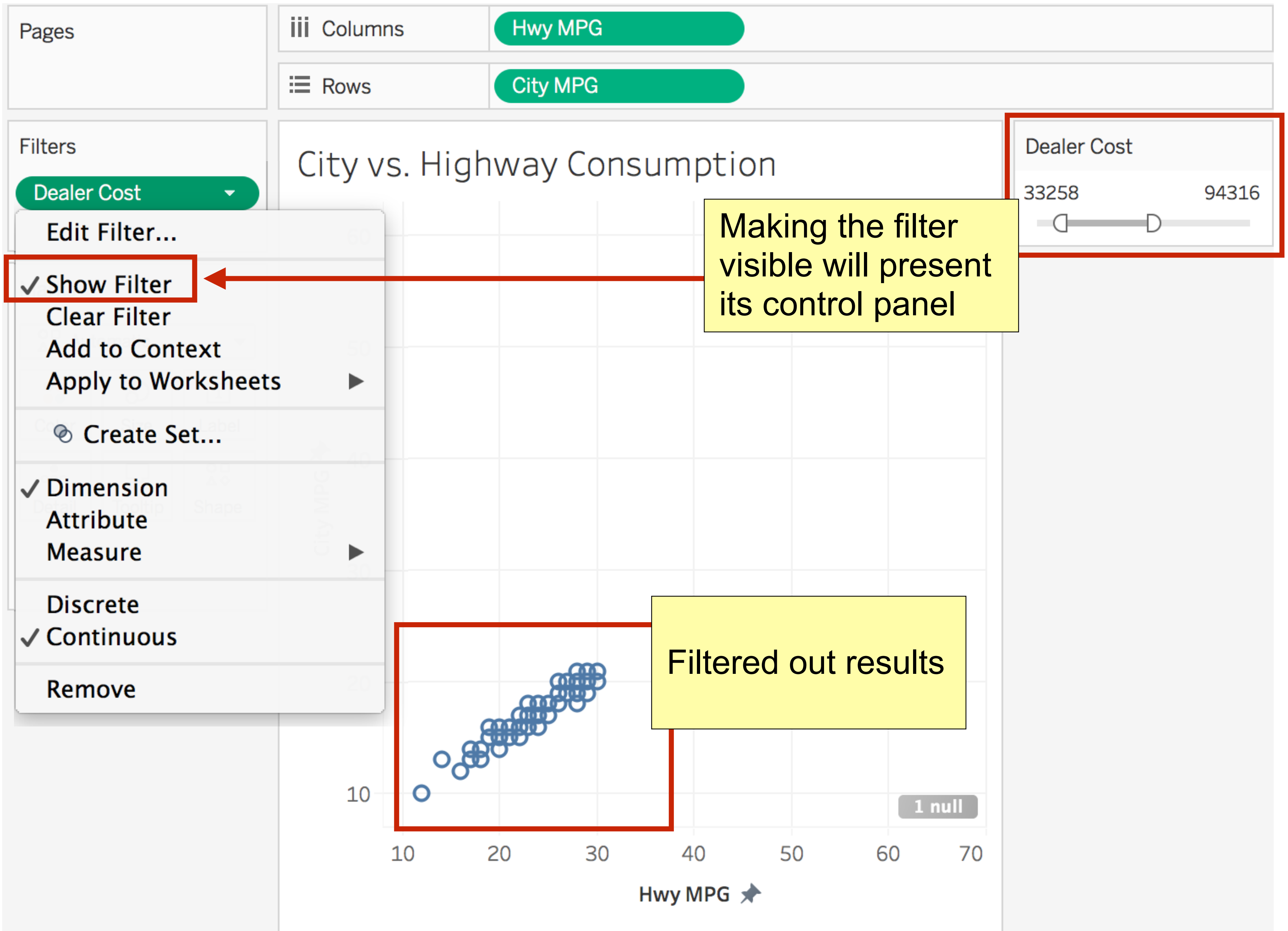
Filters: working with Continuous

We can specify how the filter will behave

Limits are extracted from the dataset

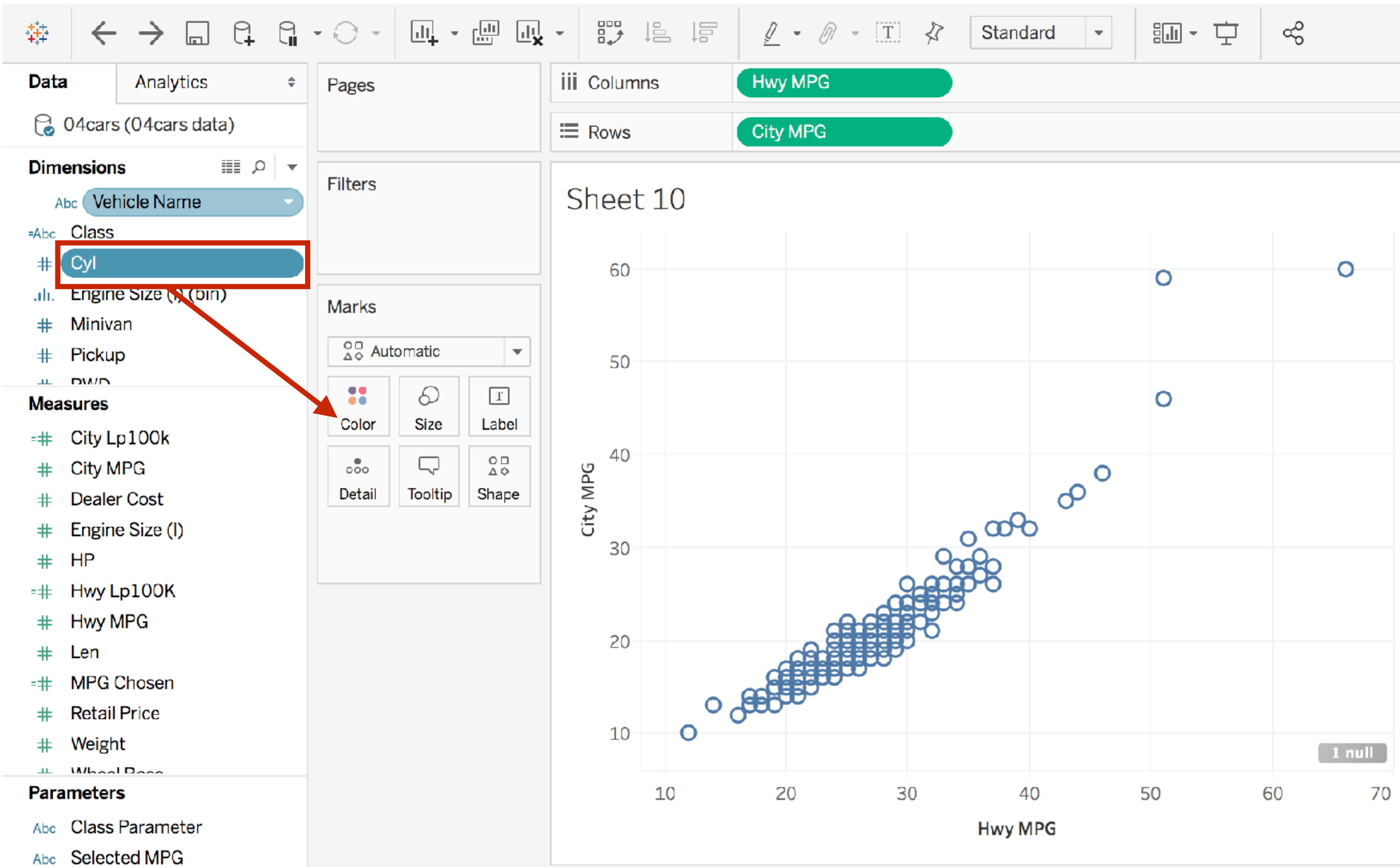


Filters: working with Continuous



Color: working with Discrete

- **Dragging a Discrete field to the Color Mark**



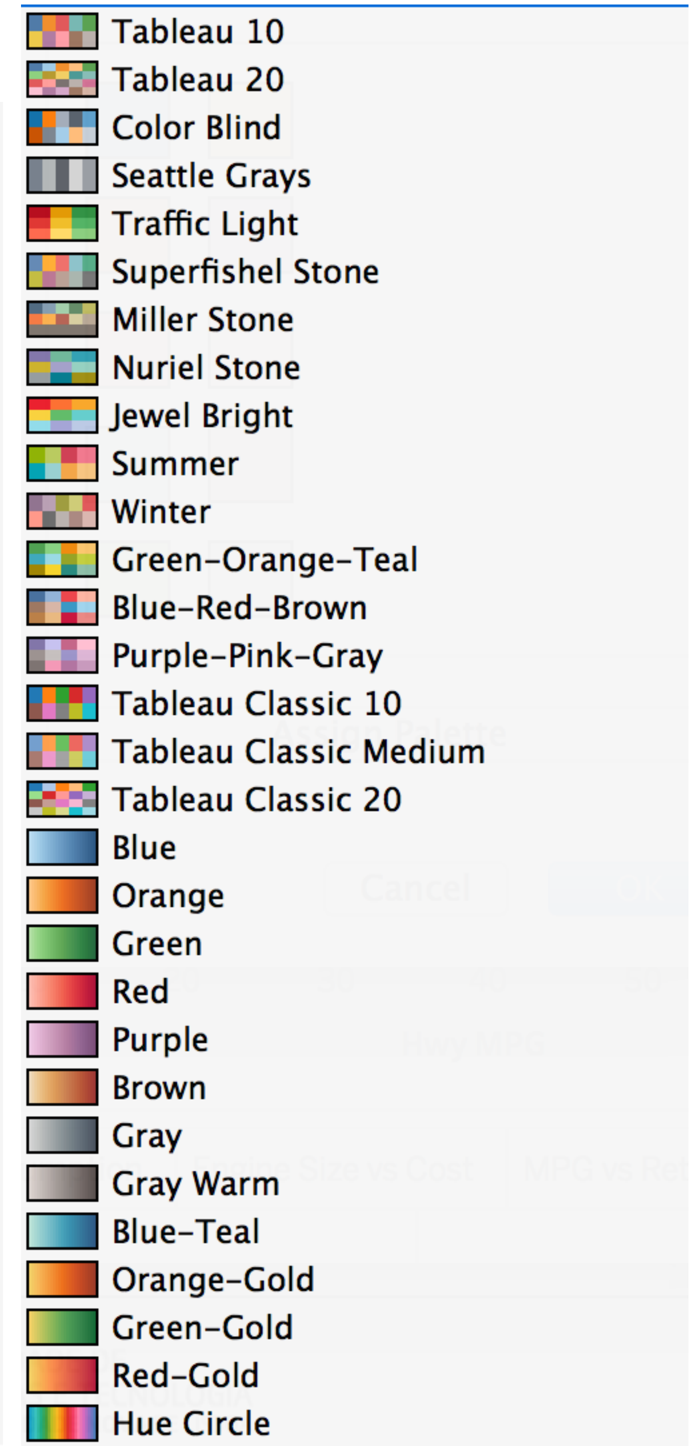
Color: working with Discrete

- **Dragging a Discrete field to the Color Mark**

For categorical variables a color palette with easy distinguishable colors is chosen.

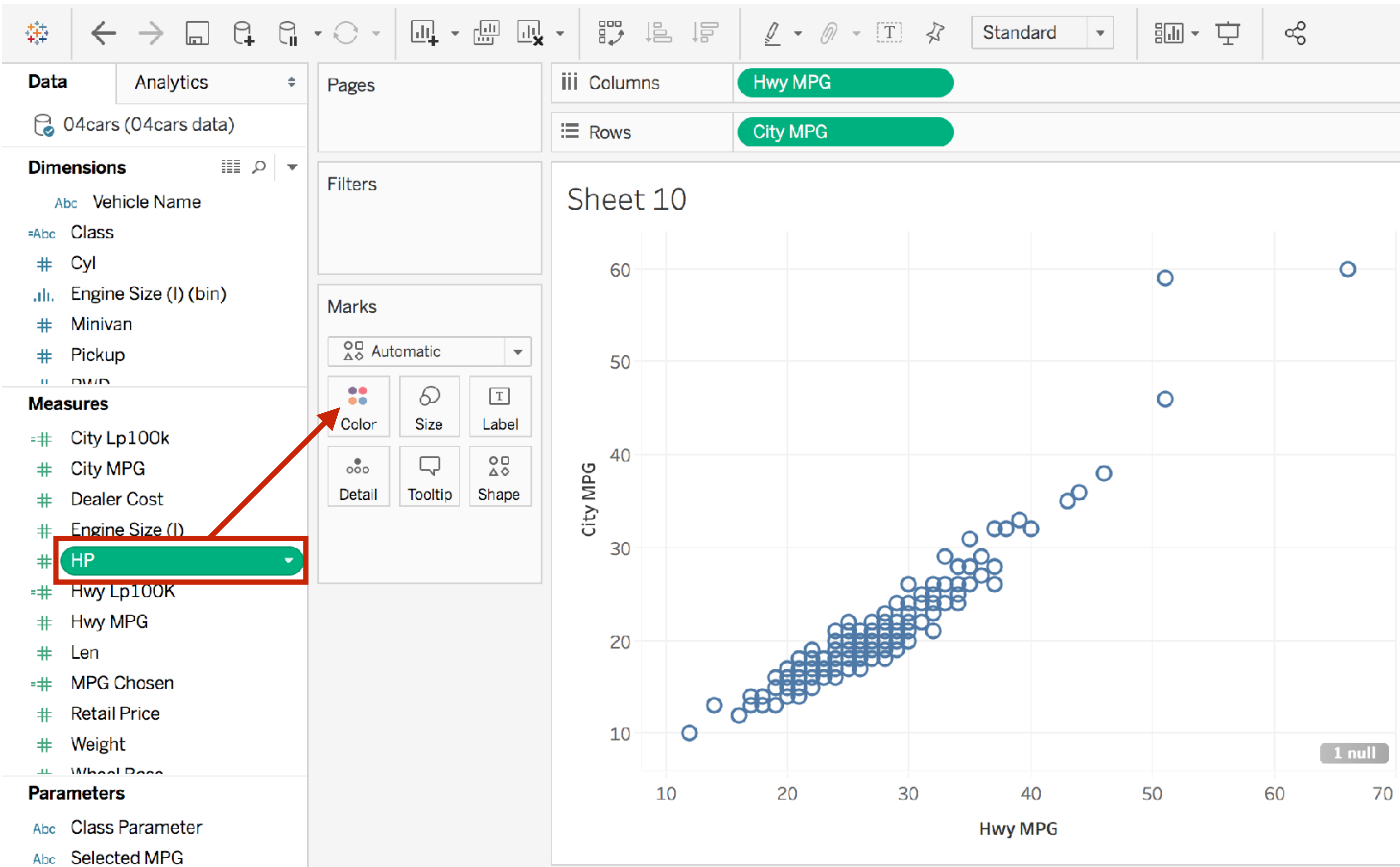
Color: working with Discrete

- **Dragging a Discrete field to the Color Mark**



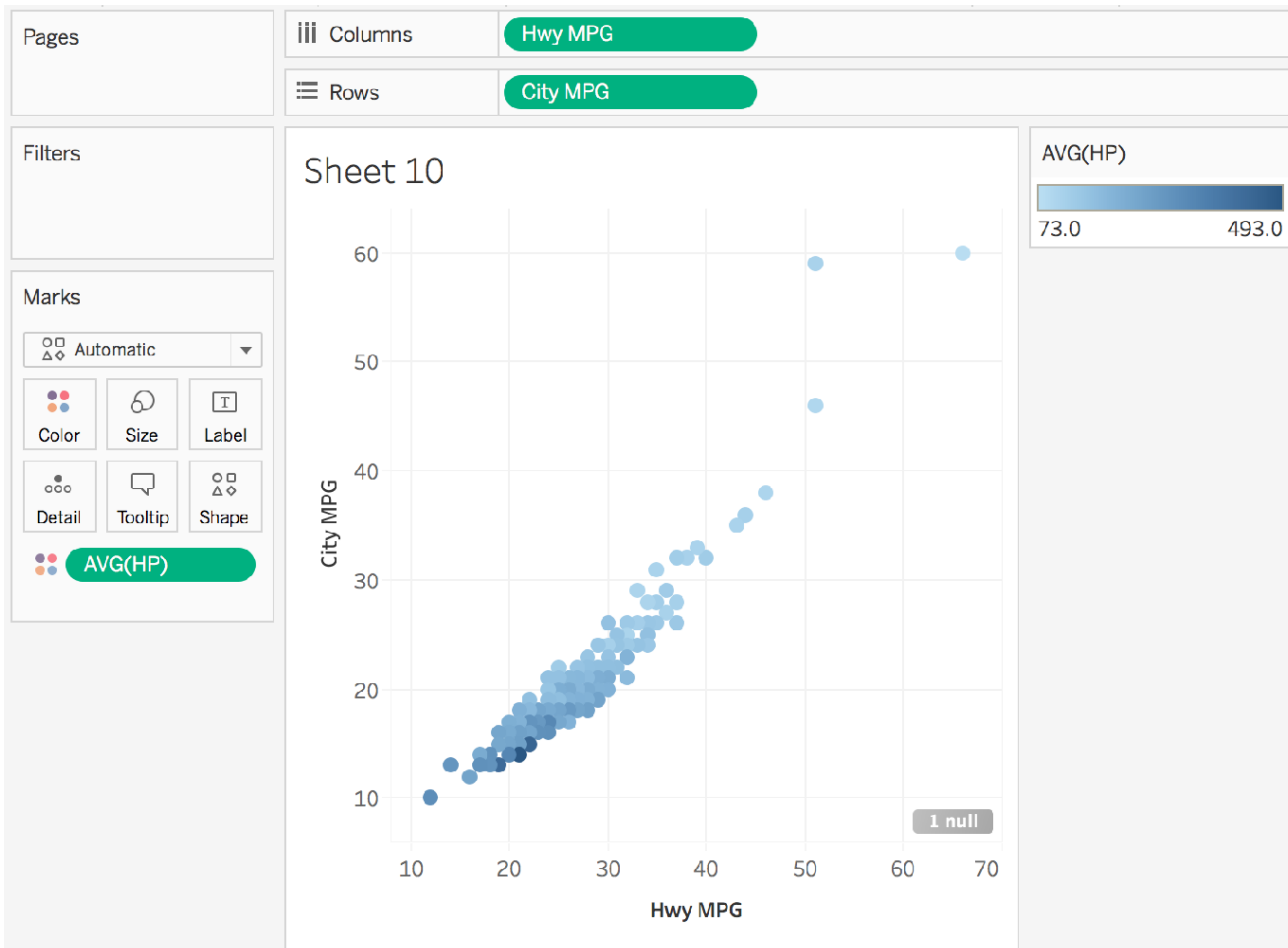
Color: working with Continuous

- **Dragging a Continuous field to the Color Mark…**



Working with Continuous (in Colors)

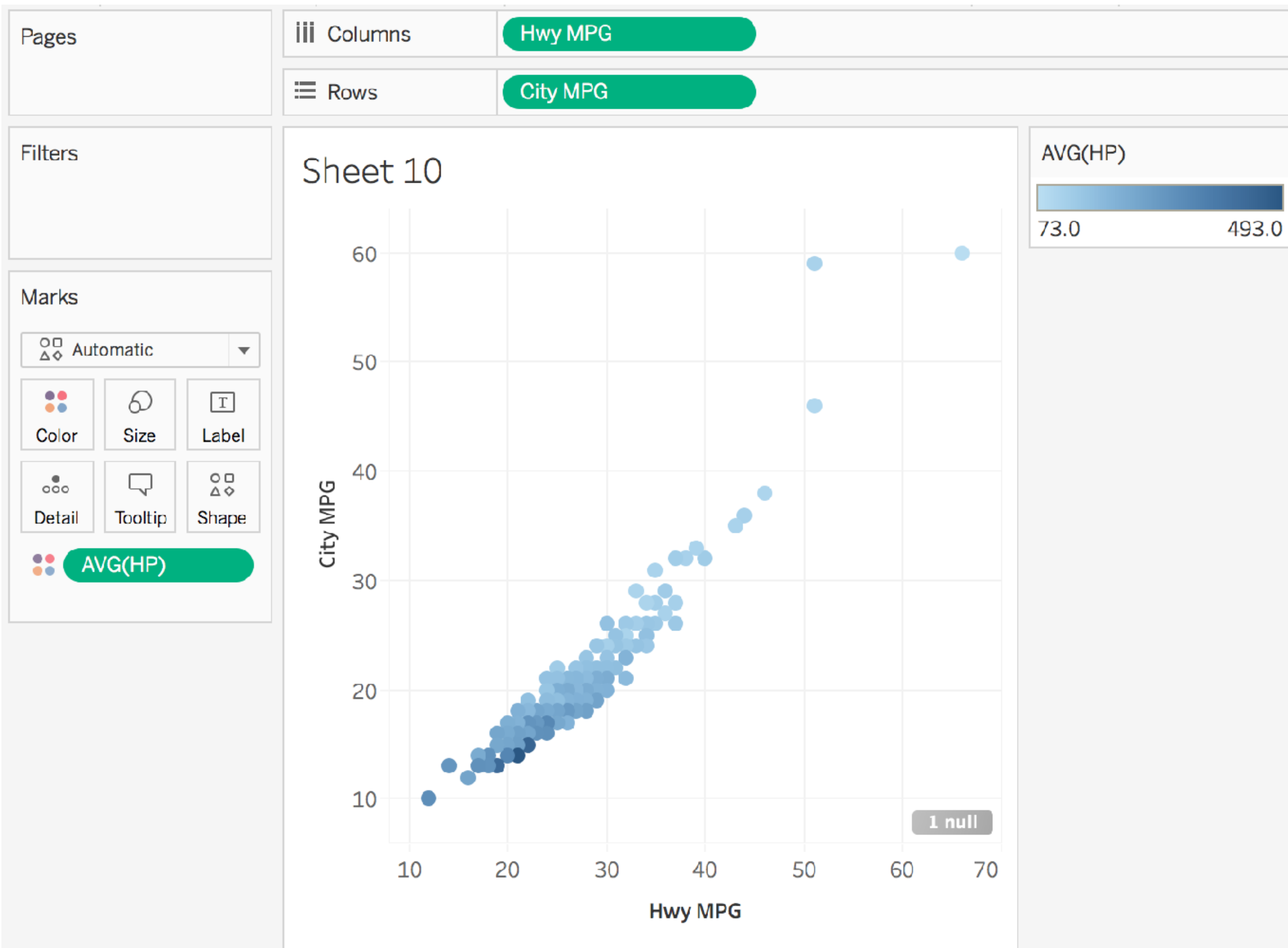
- **Dragging a Continuous field to the Color Mark**



FACULDADE DE CIÊNCIAS E TECNOLOGIA UNIVERSIDADE NOVA DE LISBOA

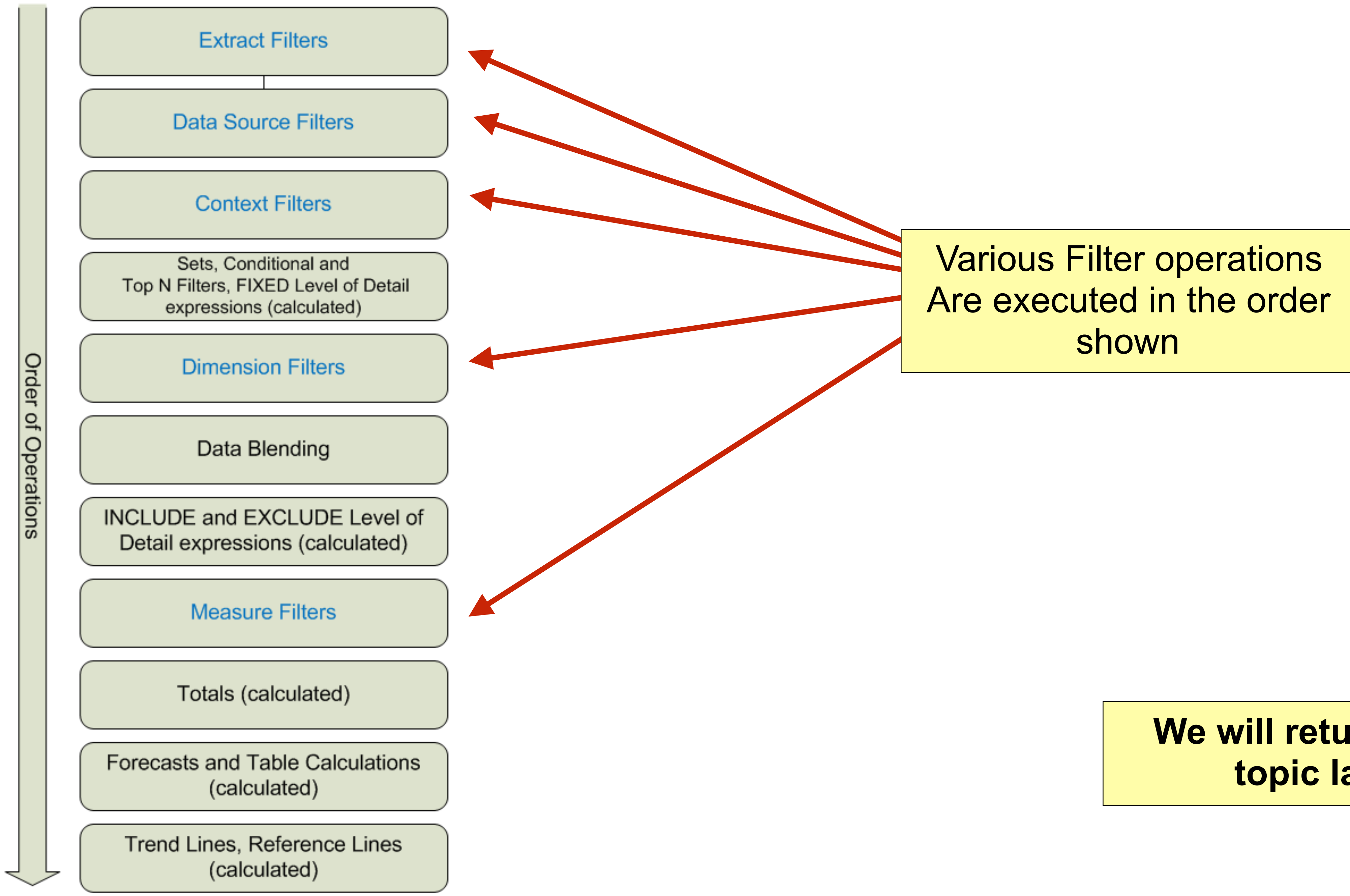
Working with Continuous (in Colors)

- **Dragging a Continuous field to the Color Mark**



Tableau's Pipeline

- **Tableau executes its operations in a pipeline, thus fixing their relative order**



We will return to this topic later !

https://onlinehelp.tableau.com/current/pro/desktop/en-us/order_of_operations.html

Further Reading and Summary

Q&A

What you should know

What you should know

- **How to import/connect to datasets**

What you should know

- **How to import/connect to datasets**
- **What are the datatypes supported and what they are usually used for**

What you should know

- **How to import/connect to datasets**
- **What are the datatypes supported and what they are usually used for**
- **What is the difference between a Dimension and a Measure**

What you should know

- **How to import/connect to datasets**
- **What are the datatypes supported and what they are usually used for**
- **What is the difference between a Dimension and a Measure**
  - Tableau aggregates measures for each dimension value (by default)

What you should know

- **How to import/connect to datasets**
- **What are the datatypes supported and what they are usually used for**
- **What is the difference between a Dimension and a Measure**
  - Tableau aggregates measures for each dimension value (by default)
- **What is the distinction between a Discrete and a Continuous field**

What you should know

- **How to import/connect to datasets**
- **What are the datatypes supported and what they are usually used for**
- **What is the difference between a Dimension and a Measure**
  - Tableau aggregates measures for each dimension value (by default)
- **What is the distinction between a Discrete and a Continuous field**
- **How tableau treats Discrete and Continuous fields…**

What you should know

- **How to import/connect to datasets**
- **What are the datatypes supported and what they are usually used for**
- **What is the difference between a Dimension and a Measure**
  - Tableau aggregates measures for each dimension value (by default)
- **What is the distinction between a Discrete and a Continuous field**
- **How tableau treats Discrete and Continuous fields…**
  - … in Rows/Columns shelves to produce headings and axis that will determine position

What you should know

- **How to import/connect to datasets**
- **What are the datatypes supported and what they are usually used for**
- **What is the difference between a Dimension and a Measure**
  - Tableau aggregates measures for each dimension value (by default)
- **What is the distinction between a Discrete and a Continuous field**
- **How tableau treats Discrete and Continuous fields…**
  - … in Rows/Columns shelves to produce headings and axis that will determine position
  - … in Filters to produce lists of values or ranges to select from

What you should know

- **How to import/connect to datasets**
- **What are the datatypes supported and what they are usually used for**
- **What is the difference between a Dimension and a Measure**
  - Tableau aggregates measures for each dimension value (by default)
- **What is the distinction between a Discrete and a Continuous field**
- **How tableau treats Discrete and Continuous fields…**
  - … in Rows/Columns shelves to produce headings and axis that will determine position
  - … in Filters to produce lists of values or ranges to select from
  - … in Color to produce palettes of colors to quickly distinguish different values or a continuous scale to compare values

Recommended Activities

- **See again the video on <u>Getting Started</u>**

- **See the video on <u>Managing Metadata</u> (4 min)**

THANK YOU!

jmp@fct.unl.pt

fpb@fct.unl.pt

FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

NOVALINCS
LABORATORY FOR COMPUTER SCIENCE AND INFORMATICS

Departamento de Informática, FCT/UNL
Quinta da Torre P-2829-516
CAPARICA, Portugal

di.secretariado(AT)fct.unl.pt

(+351) 212948536 (direct)
(+351) 212948300 (central)
(+351) 212948500 (central)