DS342 - Data Analytics

Lecture 7
Data Analysis with Power Pivot

Dr. Marwa Sabry

4-3 Data Analysis with Power Pivot

- This section discusses the most publicized member of Microsoft's Power BI tools, **Power Pivot**.
  - The Data Model and Power Pivot work hand in hand for powerful data analysis.
  - Make sure Power Pivot is loaded by checking its item in the COM Add-ins list.
  - It is easy to go back and forth between the Excel window and the Power Pivot window. Both can be open at the same time.

Standard PivotTable or Data Model PivotTable?

Choice between: Standard PivotTable & Data Model PivotTable

Standard PivotTable:

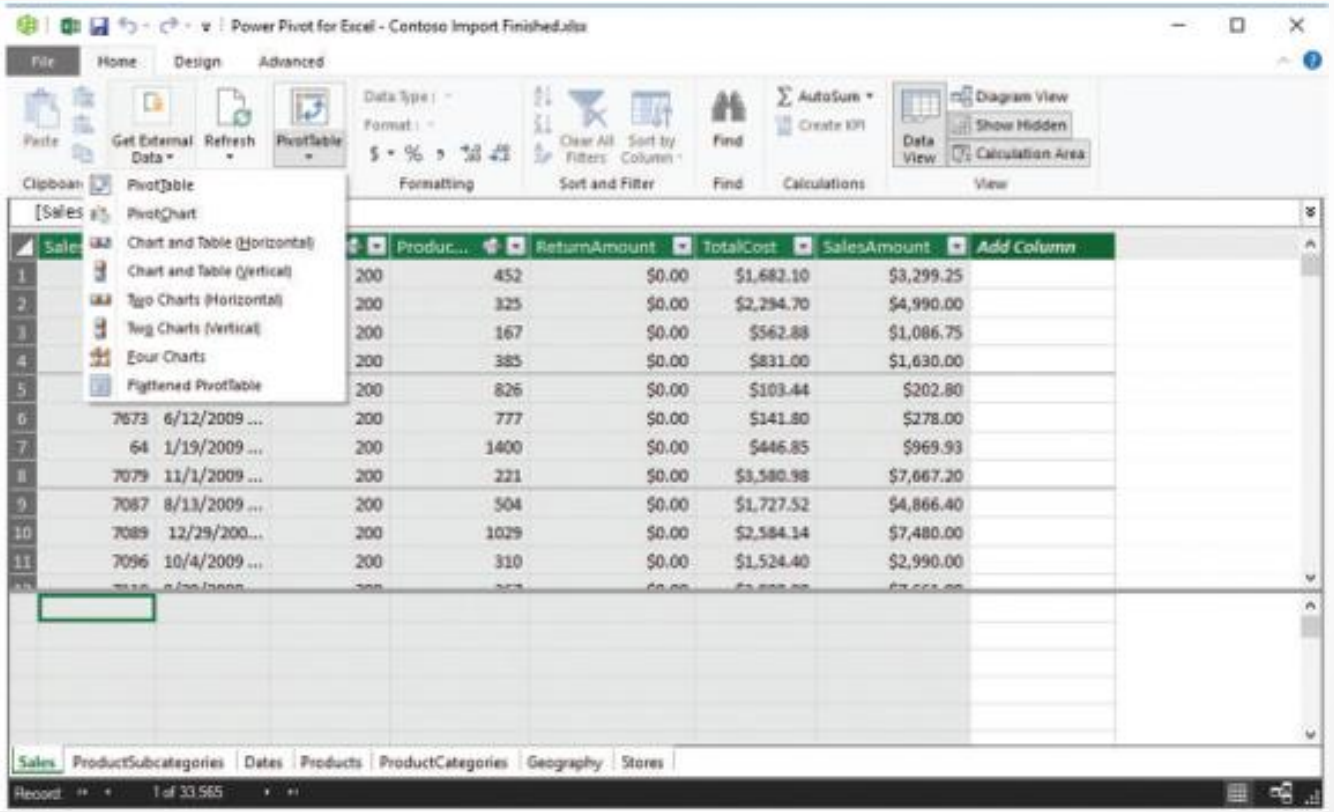
1. Have One Flat Table
2. Don't have Big Data
3. Standard Calculation in PT sufficient
4. Must manually add Number Format for each new Calculation
5. Can NOT re-use a Formula
6. For simple PivotTable Reports on a small data set, Standard PivotTables are great.

Data Model PivotTable:

1. Have Multiple Tables
2. Have Big Data
3. More Varied Calculations with DAX
4. Number Formatting can be added to formula
5. DAX Measures (Formulas) are created once, and can be re-used many times
6. For complex projects or Big Data, Data Model PivotTables are great.

4-3a Basing Pivot Tables on a Data Model

- If Power Pivot is loaded, you can click the PivotTable dropdown arrow in the Power Pivot window to see the options in this figure.



4-3b Calculated Columns, Measures, and the DAX Language (slide 1 of 2)

- The feature that sets Power Pivot apart from regular Excel pivot tables is the relatively new **Data Analysis Expressions (DAX) language** for performing calculations.
  - Two main things you can do in the Power Pivot window is add calculated columns and add measures, and you do each of these with DAX formulas.

4-3b Calculated Columns, Measures, and the DAX Language (slide 2 of 2)

- A **calculated column** is basically like a new column in Excel based on data from other columns in the selected table or even columns in related tables.

- A **measure** is a summary function of some type.

Example 4.4: Using DAX on the Contoso Data

(slide 1 of 10)

- **Objective**: To see what DAX calculated columns and measures look like and how they can be used in pivot tables.
- **Solution**: For illustration, we will create two types of calculated columns. The first is like a typical new column in Excel. We'll call it Profit, calculated as SalesAmount minus ReturnAmount minus TotalCost.

Example 4.4: Using DAX on the Contoso Data

(slide 2 of 10)

Solution:

1. Click anywhere in the right blank column, the column with the Add Column heading.

2. Type an equal sign, click the SalesAmount heading, type a minus sign, click the ReturnAmount heading, type another minus sign, click the TotalCost heading, and press Enter.

Example 4.4: Using DAX on the Contoso Data

(slide 3 of 10)

Solution:

3. The default heading for the new column is Calculated Column
    1. This is too generic, so right-click the heading, select Rename Column, and type Profit.

Alternatively, you can combine steps 2 and 3 by typing **Profit:=Sales[SalesAmount] - Sales[ReturnAmount] - Sales[TotalCost]** in the formula bar.

Example 4.4: Using DAX on the Contoso Data

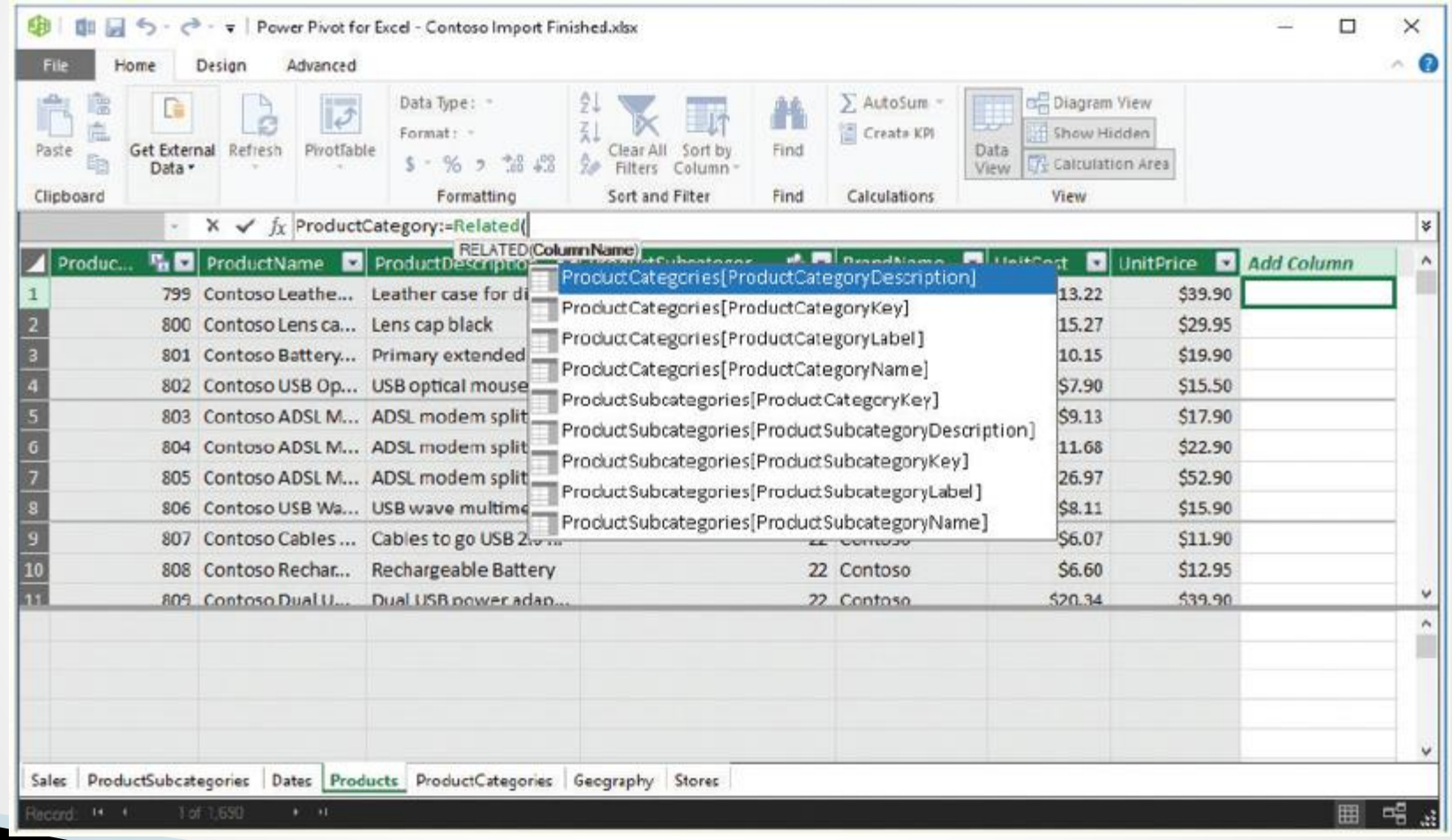
(slide 4 of 10)

- The second example of a calculated column uses a DAX function, RELATED, that has no counterpart in Excel.
- RELATED is the function that we use in DAX to look up an item from the Many Side of a Relationship to retrieve an item from the One Side of the Relationship.

- To use this function, select the Products table, click anywhere in the first blank column, and type **ProductCategory:=RELATED**.

Example 4.4: Using DAX on the Contoso Data

(slide 5 of 10)



Example 4.4: Using DAX on the Contoso Data

(slide 6 of 10)

- Why would you want to have information on product category and product subcategory in the Products table, given that this information already exists in two related tables?
  - First, you can now hide the ProductCategories and ProductSubcategories tables.
  - Then, when you use the Data Model to create pivot tables, all information about products will be in a single Products table, which is less confusing to a user.
  - The second reason is that now you can build a **hierarchy** in the Products table.

Example 4.4: Using DAX on the Contoso Data

(slide 7 of 10)

- Do this as follows:
  1. Switch to diagram view in the Power Pivot window.
  2. Move the cursor over the Products table. You will see two buttons at the top right, Create Hierarchy and Maximize. Click Maximize to expand the table and then click Create Hierarchy.
  3. You will see a new Hierarchy field in the list, which you can rename Product Structure. Then drag the fields ProductCategory, ProductSubcategory, BrandName, and Product-Name, in that order, slightly below the hierarchy name, each slightly below the previous field.

Example 4.4: Using DAX on the Contoso Data

(slide 8 of 10)

Example 4.4: Using DAX on the Contoso Data

(slide 9 of 10)

- We now turn to measures, which are arguably even more important than calculated columns.
- A measure is a summarization of data intended for the Values area of a pivot table.
- The flexibility of measures created by DAX is the real “power” in Power Pivot.

Example 4.4: Using DAX on the Contoso Data

(slide 10 of 10)

- Measures can also be created *explicitly*, and the only limit is your imagination and your expertise in DAX. There are two ways you can create a measure explicitly:

  1. You can type a DAX formula in the Power Pivot window. You can enter it in *any* table of the Data Model, but you normally select the table most closely associated with the measure.

  2. You can click the Measures dropdown arrow on the Power Pivot ribbon in the Excel window.

- These two items allow you to create new measures or edit existing measures.

4-3b Calculated Columns, Measures, and the DAX Language

- The pivot table functionality that has been part of Excel for decades is great, but it has its limits in terms of generating the reports that businesspeople need.
- With Power Pivot and its DAX language, these limits are expanded significantly.
- It takes a lot of work to learn and master DAX, but this work is well worth the effort.
- DAX experts are in high demand.

DAX Formula Benefits

- ✓ DAX Measure can have Number Formatting Applied.
- ✓ DAX Measures can be Re-Used.
- ✓ DAX Formulas work efficiently on Big Data.
- ✓ DAX Formulas can make more varied calculations than in a Standard PivotTable.

Overview of Steps to Build Power Pivot Data Model & Dashboard

1) Use Power Query to Extract & Transform Data and Load to Power Pivot Columnar Database in the Data Model
2) Create Relationships between Tables
3) Create DAX Formulas: 1) DAX Calculated Columns & 2) DAX Measures
4) Hiding Fields or Tables from Client Tool
5) Refine Data Model in Power Query
6) Create Date Table
7) Create Reports and Dashboard

Excel Power Pivot & Power BI Desktop?

Choice between: Excel Power Pivot & Power BI Desktop

Excel Power Pivot:

1. Power Query, Columnar Database, Relationships, DAX Formulas are almost identical in both.
2. PivotTable Report is what you want
3. Have Excel Worksheets to compliment Data Model PivotTable Reports that allow you freedom to:

a. Work in cells, not columns and tables
b. Have any of the other Excel features to compliment Data Model PivotTables

4. Familiar with Excel.
5. DAX Formula calculate more slowly in Excel because they are calculated with MDX, which uses only one processor at a time.
6. Hard to share Power Pivot Report.

Power BI Desktop:

1. Power Query, Columnar Database, Relationships, DAX Formulas are almost identical in both.
2. More varied Visualizations and Reports
3. Visualizations and Reports are interactive (one can filter the other)
4. You can publish Visualizations and Reports, so they can be consumed on any device.
5. Table DAX Formulas can be part of the Data Model as a Table.
6. DAX Formulas calculate more quickly in Power BI because they are calculated using DAX which allows parallel processors to work on calculations. This matters for big data.
7. PBID gets updates each month & sometimes we get DAX Formulas or other features that aren't in Excel (why we can't create Data Model in PBID & open in Excel, but reverse is possible).