MASTER

Visual Analysis of Revenue Key Performance Indicators

Vazir, Aanchal P.

Award date:
2020

Department of Mathematics and Computer Science
Visualization

Visual Analysis of Revenue Key Performance Indicators

Master Thesis

Aanchal Vazir

Supervisors:
Huub van de Wetering, TU/e
Siebert Looije, FrieslandCampina

Eindhoven, 2020

Abstract

For any given company or organization, growing and escalating the business is one of the major objectives. Generally, the measures against which companies measure and evaluate their performance or success are known as Key Performance Indicators (KPI)'s. While the higher-level business targets are focused about "generating revenue" and "increasing profits", organizations also need to think about balancing such profitability-oriented targets against sales objectives for devising their business strategies. In theory, increasing product's price can help companies realize higher profits but this may bring down the demand and thereby, sales of the product. This trade-off between sales and profitability is crucial for businesses to account for, to determine their product's pricing. This requires, among other aspects, the use of strong analytics, and understanding the impacts of price change on a series of dependent measures. In order to guide pricing decisions and enhance the understanding of the underlying parameter space, visualization can be used to augment human capabilities such as perception, comprehension and decision making. As a realistic use case, we present a visual analysis tool that can help business stakeholders with pricing decisions by modeling and computing the impact of price change of products on a set of KPI's, using powerful visualization techniques. The process of designing and developing this visual analysis tool includes determining user tasks or actions to help them achieve their visualization goals. On conducting a a controlled user study with 7 of the potential end users, we received qualitative feedback about the usability of the developed tool.

Preface

This master's project report marks the culmination of my studies at the Technical University of Eindhoven for obtaining a Master of Science (MSc) degree in Data Science in Engineering (track of Computer Science and Engineering). The graduation experience was quite unusual and unexpected due to the unfortunate corona crisis. Nevertheless, there was never a lack of motivation to pursue this project that helped me learn how to bridge the gap between business oriented decision making, and data analysis and visualization. While the thesis assignment helped me chase my passion of exploring the data visualization domain of data science, it gave me an opportunity to use my knowledge for developing a practical use case. The successful completion of this project would not have been possible without the constant support and guidance from several people. First of all, I would like to thank FrieslandCampina for providing me with this opportunity of undertaking a master thesis assignment with their Digital Media and Insights team. They ensured that I was provided with all the necessary resources to conduct my research and implementation. I would like to thank the team and my fellow colleagues for constantly supporting me through brainstorming and feedback sessions, and maintaining virtual contact in these strange times.

I would like to thank Siebert Looije, my daily supervisor, for always finding time to guide me through every checkpoint of the project. His knowledge and expertise in the business domain helped me gain broader perspective of the project and its usefulness. I am thankful to him for devoting his time to steer and motivate me towards making the right choices and decisions in moments of confusion. I would also like to extend my sincere gratitude to Huub van de Wetering, my supervisor at TU Eindhoven, for shaping my approaches and thought processes throughout the project. His ideas improved my approach towards achieving results. I could improve the quality of my work with the help of his recommendations. I would also like to thank the member of the jury, Dirk Fahland for agreeing to be a part of my thesis committee. I would like to use this opportunity to thank Tjadi Peeters for helping me with the implementation whenever I was stuck. Special thanks to Arjen Brouwer, Rui Conde, Athith Rai, Oeljana Smits, Andrea Kuijt, Maxime Egdom, Bram Verleur, Khiet Anh Nguyen, Kasper Birkelund, Arjen van den Heuvel, Janno Bannink, and Carolien van Alphen for helping and motivating me in several different ways throughout the course of my graduation internship.

Finally, I would like to thank my parents and sister who believed in me and stood by me during the entire course of my study in the Netherlands. Without their support, I would not have been able to accomplish this work. Last but not the least, I would like to thank my friends and well wishers for their support and encouragement to face the challenges of this project.

Contents

Chapter 1

Introduction

This chapter introduces the reader to the company for which the thesis project was undertaken, the scope of the problem and how visualization can help us solve it, the contributions of our work and an overview of the report.

1.1 About the company

FrieslandCampina is a Dutch multinational Dairy Cooperative which is based in Amersfoort, The Netherlands. It supplies consumer products, such as milk, yogurt, cheese, infant nutrition and desserts, products for the professional market, such as cream and butter products, ingredients and semi-finished products for producers of infant nutrition, the food industry and the pharmaceutical sector. The company is wholly owned by Zuivelcoöperatie FrieslandCampina U.A.(*Uitgesloten aansprakelijkheid*, literally translated as "excluded liability") with 17,413 member dairy farmers in The Netherlands, Belgium and Germany. This means that, via the cooperative, the company's member dairy farmers are the joint owners and supervisors of the company. FrieslandCampina has branches in 36 countries and they export to more than one hundred countries worldwide. The company's activities are divided into four market-oriented business groups; Consumer Dairy, Dairy Essentials, Ingredients, and Specialized Nutrition[5].

1.2 Scope of the problem

The project concerns the Consumer Dairy business group of FrieslandCampina. Once every year, FrieslandCampina and the Retailers selling their products, sit together to fix the price of products. The outcome of such a price strategy must put both parties in a rewarding position in terms of their profit margins. An organization's pricing decision can be strategic and/or tactical. A strategic approach aims at defining long-term goals and planning a road map of how to achieve them. Tactics are rather concrete, short-term measures that are oriented toward taking smaller steps within a shorter time frame. The scope of this project is to address and answer tactical questions with regards to the price of products; what would happen to the Sales Volume and Margin, if the price of product(s) is changed at the moment? Thus, the problem can be stated more formally as the following:

If the price of a product is increased or decreased, what is the impact on the four major Revenue Key Performance Indicators (KPI's):

1. Sales Value
2. Sales Volume
3. Net Sales Value

4. Margin

For every organization and its business, there are a certain set of metrics or measures against which the success, performance, growth and potential of the business can be quantified or measured. These are termed as KPI's. These aspects are discussed in more detail in section 3.4 and throughout this report, we will refer to these in context of FrieslandCampina. The objective of the thesis project is to learn the impact of a momentary price change of product(s) on the Revenue KPI's and show relevant insights.

An effective and efficient alternative of cognitive calculations of revenue KPI's is visual representation of data for extracting and communicating insights. It is effective because visual representations have the ability to show-and-tell, and efficient because they can bolster the process of comprehension and decision making. The chances of inferring more from data are high when visual methods are used for portraying trends and patterns that may otherwise be hidden behind the numbers. Taking our scope into consideration, the problem is more than just visualizing trends, patterns or behavior. There arises a need of observing changes, impacts and outcome of change in a variable on a set of dependent variables.

To imitate operations and processes of a system, subject or a concept, simulators are often employed. To get a feel of how price impacts our KPI's of concern, a simulation environment would help us recreate different scenarios in a realistic manner, to understand how the parameter space behaves with respect to change in price of products. Hence, the development of a simulator is chosen as one of the possible solutions for providing a framework that models all the variables and parameters of concern. Visualization can enrich a simulator for making data more appealing and analysis more intuitive. However, to make room for exploratory analyses where the outcome of the pursued user activity is not necessarily known at the beginning, the end-user must have control over the tool to specify what he or she is looking for and guide their own process of analysis. The project's aim is to build such a simulator as a visual analysis tool to support user interaction for extracting knowledge in the domain. As a proof of concept, the work shown in this report discusses how visualization can help make tactical pricing decisions.

1.3 Contributions

The visual simulator developed provides dairy businesses with a framework to measure their prime Revenue KPI's. The visual representation of these performance indicators helps higher-level strategic users as well as other stakeholders such as managers and business analysts draw insights on how their business is performing with their current pricing strategy and visualize how these KPI's would be impacted over time on changing their pricing tactics (either by increasing or decreasing their current price of products). Questions like how to visualize the data and what questions do the users expect the visualizations to answer, are addressed. An implementation of the user-centric visual analysis tool through iterative improvements, is shown. An evaluation of the end product conducted through a user feedback test helps in assessing and identifying how the project can be scaled up for deployment.

1.4 Overview of the report

The related work with regards to our project's description and implementation is given in chapter 2 of this report. The required background knowledge of the underlying concepts and theories have been explained in chapter 3. Next, a formal and deeper outlook to our problem in the form of user questions and the first step taken in approaching our problem's scope is discussed in the chapter 4 of this report. Further, details about the requirements and constituents of the user interface to be developed, are narrated in the chapter 5. Apart from that, a description of the available data and formulation of user tasks are also elaborated upon in this chapter. In the subsequent chapter 6 of the report, the chosen design principles to produce visualizations with the given data and user tasks are explored in depth. Based on these design principles, the implementation and results of the visual analysis tool are described vividly in chapter 7. In chapter 8, evaluation of the developed tool based on our research questions and a user test conducted, has been discussed. Finally, we conclude our work by stating what we have been able to achieve, proposing some recommendations and outlining the future scope of the project in chapter 9 of the document.

Chapter 2

Related Work

The field of data visualization continues to expand with ongoing research being conducted by people. Creating visualizations requires making several decisions and judgements. For solving any given problem, general approaches involve understanding of the context, importance, dependencies, feasibility, benefits, and drawbacks pertaining to the problem. Once these aspects are investigated, the next step is to try and solve the problem based on numerous findings. For conceptual or theoretical problems, such as studying the price elasticity of demand of products, the support of visual methods can be an effective means of transfer of knowledge to both expert or non-expert users. In order to create visualizations, one must determine which questions to ask, identify data properties, and select the appropriate visual encoding techniques to map the data attributes with values. Jeffrey Heer et al.[7] in their work, provide a brief tour through the "visualization zoo" showing several techniques for producing interactive visualizations with diverse data sets. They focus on a couple of sophisticated techniques for time series data sets among others. In an interesting argument, authors Hans-Jörg Schulz et al.[18], state that Information Visualization is not only about visual representation of data for the reinforcement of human cognition. It also takes into account the tasks or activities that the user wishes to pursue with it. In their work, they show how knowledge about these tasks plays an important role in choosing or creating suitable visual representations. Through the dimensions of their design space, they characterize the main aspects of tasks and explain them with the help of a use case.

In contemporary business organizations, where business intelligence and analytics has emerged as an important area of study and utility, analytics is used for objectives such as discovery, interpretation, and communication of patterns in data. This further enables organizations to dynamically render data with a variety of visual, interactive representations. Among numerous other application domains, analytics can lead the way on pricing and customer profitability for businesses. Larry Montan et al.[2] claim that by setting prices according to the mathematics handed over by the pricing analytics team of an organization, the revenues do surely increase. The authors write about how there is a considerable room for improvement in determining pricing and recommend combining pricing with analytics, to create a mechanism that functions as both a catalyst and a metrics engine for managing profitability. However, to aid decision making processes for subjects such as pricing, visualization can be combined with analytics. Visual analytics is based on the principle of employing practices such as data mining and statistical work to visualize information in a format that is easy for humans to understand. In their study, authors Enamul Hoque et al.[9] state that "a well-accepted principle in visual analytics is the need to support interactive exploration and iterative view refinement". They imply that developing an interface that helps users answer the questions they have about their data, can help them attach analytical reasoning behind their decisions.

While organizations tend to use existing Business Intelligence Tools and software for faster and easy reporting, analysis and planning, recently, usage of rich frameworks and platforms with built-in libraries, requiring minimal programming expertise from the developers, has become popular. One of these is the Dash framework by plotly, that lets python programmers develop interactive, customized and high-end web applications without the explicit use of scripting languages like Javascript. Dash provides a Python interface to a rich set of interactive web-based components. Some of the freely accessible dashboards for financial reporting represent the framework's usability for web application development[17].

Creating user-centric applications requires gathering user requirements and tasks to provide suitable functionalities. One of the oldest and most important principles proposed by B. Shneiderman and A. Aris [19] is that information visualization designers must keep the visualization procedure in mind; *Overview* first, then *zoom* and *filter*, and view *details on demand* to enrich the analysis process for the user. One of the common user tasks in data analysis is visual comparison. In their work, Michael Gleicher et al.[6] focus on problems such as comparing two different graphs. They propose assisting visual comparison through *Juxtaposition*, *Superposition* and *explicit representation of the relationships* methods. In line with the objective of comparing graphs to observe correlations, similarity, or deviations, the technique of displaying a series of graphics using the same scale and axes, termed as small multiples, is used. Small multiples are applied to data analysis in different application domains. For instance, Aran Lunzer et al.[12] in their work, use small multiples for comparing cancer simulation results within a large parameter space.

For the development of an interactive user application that majorly relies on visualization of data relationships by defining user tasks, the motivation for characterizing questions like WHAT, WHY and HOW in context of a visual tool's design, has been gathered from the concepts laid in Tamara Munzner's work[13].

By studying the pool of related literature in the domain of pricing as well as information visualization techniques, our work tries to incorporate these concepts to produce an interactive visual analysis tool for the underlying scope of this project.

Chapter 3

Background

This chapter focuses on several concepts that provide some background knowledge on the project topic as a prerequisite to understand the project's objectives.

3.1 Sales flow of FrieslandCampina products

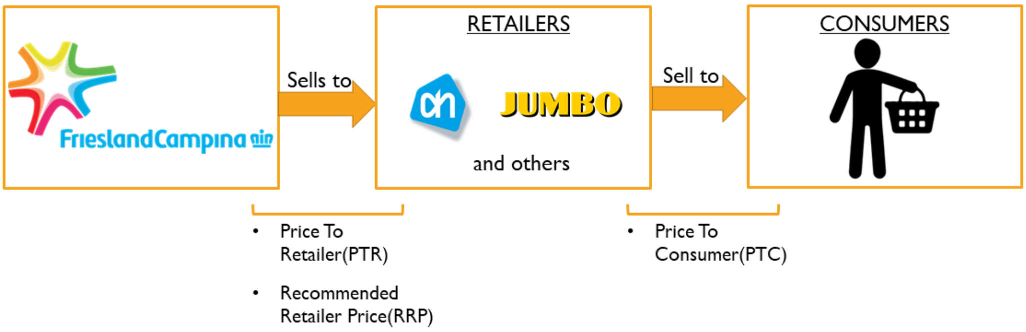


Figure 3.1: Sales flow of products from FrieslandCampina to consumers through retailers, at their respective prices.

As a dairy cooperative also owned by farmers, the company's business is to make sales of *all* the milk supplied by the farmers in form of different forms and products. It is important to emphasize on *all* the milk because ideally, all the quantity or volume of the yield must be ultimately sold. A typical sales flow of all the dairy products is illustrated in figure 3.1 and can be described as follows. FrieslandCampina sells its products to Retailers at a price termed as the *Price to Retailer* (PTR). At the same time, a *Recommended Retailer Price* (RRP) is the price that FrieslandCampina advises the retailers to sell their products at. FrieslandCampina can only advice prices and certainly not force it upon the retailers since is it illegal. Irrespective of the advised Recommended Retailer Price (RRP), the retailers sell these products to the consumers at a price termed as the *Price to Consumer* (PTC).

3.2 Product hierarchy classification

FrieslandCampina sells diversified products in the market. These product items are combined under a particular brand and further under a particular category. These items may be offered in different sizes, flavors or packages. The entire product chain comprises of a product hierarchy which can be classified as the following:

1. Category
2. Segment
3. Manufacturer
4. Brand
5. Flavor
6. Package
7. Pack size
8. Product item

In the scope of this project, we shall consider product clusters instead of individual product items. A product-cluster is a group of products which share the same price and promotion strategy. Here, promotion strategy refers to the strategy for selling products on incentives or special discounts and offers for a definite period of time. A product hierarchy classification can be understood better with an example. Let's take *FrieslandCampina* as the Manufacturer, then in the *Drinking Yoghurts* Category, one of the home brands is *Optimel*. Flavors could be categorized as *Vanilla*, *Natural* and others which are sold to consumers in different packages and pack sizes depending on the product. Hence, different flavors and a defined pack size of *Optimel*'s *Drinking Yoghurts* could be grouped together as a product-cluster.

Now that these terminologies have been defined, we can state the scope of the project mentioned in section 1.2 more specifically as the following:

If the price of a product per Category, Brand or Retailer is increased or decreased, we are interested in gaining insights of its impact on the four major Revenue KPI's; Sales Volume, Sales Value, Net Sales Value and Margins.

3.3 Revenue growth management

One of the four P's of marketing, Price, has great power to transform a company's revenue and profits (the other three are Product, Promotion and Place). In fact, Pricing is considered one of the most powerful levers for improving profitability[3]. At FrieslandCampina, Pricing and Promotion are topics that fall under the so-called Revenue Management domain. This thesis assignment centers around the Pricing pillar. Both in theory and practice, it has been observed that an increase in price has the greatest relative impact on profit[16], but there are numerous key challenges involved in realizing these impacts; it requires, among others, strong analytics, combining different data sources and identifying the complete chain of dependent variables. One of the main areas of how companies determine pricing to set and enforce profitability of their products is *Strategy*. A pricing strategy encompasses the guiding principles behind product pricing. There can be two approaches to pricing; strategic and tactical. While strategic pricing focuses on long-term goals like maximizing long-term profitability, tactical pricing lays its focus on short-term objectives. Recently, the need and applications of pricing analytics are being discussed by researchers or organizations[1][2]. By combining pricing and analytics, a mechanism for managing and improving revenue related strategies can be created. Most importantly, through analytics, the internal as well as external factors affecting profitability driven by change in price can be understood at a granular level[3]. In brief, Pricing Analytics involves looking at historical data or past transactions in order to understand profitability better. By using historical sales data with mathematical optimization, prices can be set and updated in order to maximize profit[1]. The advantage of using this is to support and develop more informed profit-boosting strategies such as changing pricing policies or adjusting prices upward or downward. However, at FrieslandCampina, while profitability is an important metric from a financial perspective, achieving sales objectives by measuring the Sales

Volume (the total quantity of products sold), is equally important for undertaking techniques like pricing analytics. This is because, FrieslandCampina is co-owned by farmers. All the volume of milk produced by the farmers must be sold, which otherwise would simply be waste. This motivates the need of developing a visual simulator for tactical pricing (referred to as a price simulator further in the report) to motivate and guide decision making in terms of pricing. However, like mentioned earlier, it is first important to understand the complete chain of dependent variables that drive the change in profitability and sales volume, when the price of products is changed. We get a closer look at it in section 4.2.

3.4 Revenue KPI's

As described earlier, KPI's are measures that help businesses understand their product performance in different markets. Since the objective of the project concerns pricing and profitability, we take a detailed look at the relevant ones.

Let us first take an example situation. Let us assume that FrieslandCampina decides to change the price of one of their products, in particular let us consider an increase in the base price of a certain product. While this price increase could in principle increase their margin, it is trivial to see that the volume demanded by retailers and thereby consumers might not necessarily increase. This means that change in price produces counter impacts on the two most important measures of the business, sales volume and margin. We state this as a hypothesis for our problem description. This explains why pricing is described as a goal-oriented procedure; whether the goal is to boost profit margin ignoring effects on volume, or focus on achieving volume, ignoring the effect on overall profitability. Research also suggests frameworks for integrating both these goals simultaneously to establish standards for making trade-offs between volume-driven pricing or profit-driven pricing[16].

The following are the four main Revenue KPI's which are taken under consideration for outlining a pricing approach:

1. Sales Volume:
   This is the quantity of the products sold(in kilograms or liters). For a dairy cooperative like FrieslandCampina, this is one of the most important business measures;

2. Sales Value:
   This is the amount earned by all retailers or stores selling the products to consumers. In other words, this is the revenue on retail sales of the products (the price that the consumers pay multiplied by the volume or quantity of the products purchased);

3. Net Sales Value:
   It constitutes the revenue generated by selling products to the Retailers. In other words, this is FrieslandCampina's revenue from the sales made to the Retailers;

4. Margin:
   This KPI is in itself a dimension comprising of a set of Margin KPI's. FrieslandCampina's margin is computed by subtracting total Cost Of Goods Sold (COGS) from the Net Sales Value. This can also be defined as the absolute earnings for FrieslandCampina.

While FrieslandCampina margin is computed by subtracting total costs from the Net Sales Value (across all retailers and products), the retailer margin is computed by subtracting the Net Sales Value (the value at which FrieslandCampina sold the products which is same as the value at which all the retailers bought them) from their total sales value (across all retailers and products). To compare the margins of FrieslandCampina and the retailers, either for a given product and brand or across all products and brands, margin pool is a metric that is used. A margin pool is the share or distribution of the absolute margins (price minus the total costs) of FrieslandCampina and its retailers. In order to obtain a deeper understanding of how change in price affects our KPI's, we now explain the concept of Price Elasticity in the following subsection.

3.5 Price elasticity

In general, people's desire towards things lessen as those things become more expensive. While for some products, even a small increase in price could cause its demand to decline sharply, for other products, the demand could remain almost the same even with a large increase in the price. Economists term this sensitivity to price increase as *elasticity*. Thus, Price Elasticity of Demand (PED) measures the responsiveness of demand after a change in price. Specifically, price elasticity gives the percentage change in quantity demanded when there is a one percent increase in price (taking all other parameters as constant)[8].

$$\text{Price Elasticity of Demand} = \frac{\text{\% Change in quantity demanded}}{\text{\% Change in price}} \tag{3.1}$$

The demand is *Price Elastic* when a change in price leads to a bigger percentage change in demand. This means that PED is greater than 1 (in formula) and a rise in price causes a decrease in the demand for the seller. A price elastic demand curve is illustrated in figure 3.2. A 10% increase in price causes an approximately 15% decline in volume. Hence, the proportion by which the volume declines is much larger than the proportion by which the price percentage is increased. The demand is said to be *Price Inelastic* when change in price leads to a smaller percentage change in volume demanded. Therefore, PED is lower than 1 (in formula) and a rise in price causes an increase in demand for the seller. A price inelastic demand curve is shown in figure 3.3. The demand here is price inelastic because a 10% increase in price causes a 5% decline in volume, a rather smaller decline percentage. The proportion of change in the x-axis variable is smaller than the proportion of change in the y-axis variable. If the price elasticity is equals to 1, the change in price causes no change in demand for the seller.

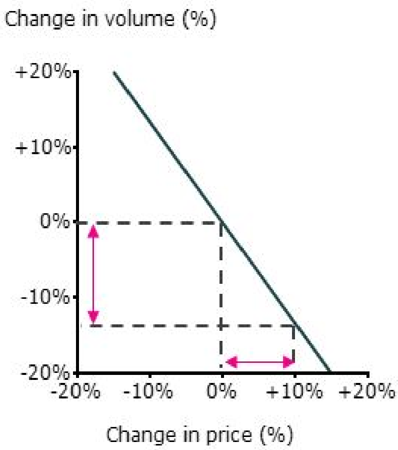


Figure 3.2: Price Elastic Demand - when price changes lead to higher opposite volume changes, i.e. a 10% increase in price leads to a 15% volume decrease.

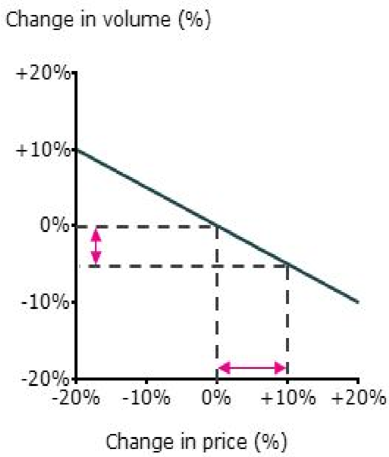


Figure 3.3: Price Inelastic Demand - when an increase in price leads to a lower decrease in the volume, i.e. a 10% increase in price leads to only a 5% volume decrease.

In order to determine pricing decisions or tactics for a given product, it is important to learn how elastic or inelastic its demand is. Based on this knowledge, volume-driven (with the goal of maximizing the sales) pricing tactics can be derived. Thus we can say that PED is one of the major dimensions of pricing decisions. This concept becomes highly relevant in our setting as well.

Pricing approaches involve a series of "what if" analyses; "what happens to the volume(quantity) if the price of the product is increased or decreased?". The scope of this project extends this "what if" analysis to FrieslandCampina's revenue and margins in response to the resulting change in the volume.

In our scope, if a product is price elastic, it means that its volume demand will drop steeper than the amount of its price increase. Let us understand this concept better through an example. Suppose our question of interest is: What would happen if we increase the price of one of our products by 10%? Although this is not immediately known and needs to be calculated with the formula 3.1, let us assume that the PED of the product under consideration is -1.2 (as per the formula this means that with 10% increase in price, there is a 12% decrease in the demand of volume). On a broader spectrum, the demand we are referring to is the demand by the consumers, but in the sales flow of FrieslandCampina's products, we saw that the products are first sold to the retailers and through them to the consumers. Hence, the demand we speak of at this point is the demand of the product by the retailers with respect to the 10% price increase. It is important to note that the PED is going to be different for each retailer as their response to the price increase would be different and this is directly going to affect the volume or quantity they will demand for purchase. Now, reacting to this price increase in the Price To Retailer (PTR), the retailers will increase their Price To Consumers (PTC). For the sake of this example, let us assume that a particular retailer further increases its price (PTC) by 10%. Let us now study how these price increases affect the KPI's. We must keep in mind, the hypothesis stated apriori (refer section 3.4); an increase in price would in principles, have a positive impact on profitability (an increase in margins) but, most likely a negative impact on sales volume.

KPI's	**current**	**new**	**impact %**
Volume	1,000	880	-12%
Net Sales Value	2,000	1,936	-3%
Sales Value (excl. VAT)	2,830	2,740	-3%
Margin FrieslandCampina	0.50	0.70	22%
Margin Retailer	0.83	0.91	10%

Table 3.1: Example Calculations on KPI's with dummy data when PTR and PTC are assumed to be increased by 10% each, given the PED is -1.2 for a particular retailer.

Prices per Liter	**current**	**new**	**impact % (assumed)**
Net Sales Value (NSV)/L	€2	€2.20	10%
Base PTC/L	€3	€3.30	10%

Table 3.2: Assuming the base prices and calculating the respective increments on 10% increase in both PTR (which is same as Net Sales Value per liter (NSV/L)) and PTC, per liter, as for this example.

As shown in the table 3.1, let us consider that the current Volume demanded or sold is 1,000 liters. Given that the PED is -1.2 and the PTR increase is 10%, using the formula 3.1:

$$-1.2 = \frac{\text{change in quantity demanded}}{10\%}$$

$$\text{change in quantity demanded} = -1.2 * 10\% = -12\%$$

Hence, the impact on the current Volume is -12%. Thus, the new volume demanded would be 12% less than the current quantity, which results in 880 liters. By making further calculations (shown in detail in Appendix A), we see how a 10% increase in price leads to a 22% increase in FrieslandCampina margins.

By making these computations, we get a deeper understanding of price elasticity of demand of a product, and how the change in price impacts the volume or quantity demanded. As the quantity of demand changes, all the variables dependent on it change accordingly. Through this example, we conclude that this product is price elastic; there is a larger decrease in volume as compared to the proportion of price increase. The example also provides a computational evidence to our hypothesis (refer section 3.4). The influence of price elasticity per retailer thus becomes very important to investigate how change in a retailer's price PTC will impact the total sales volume and thereby margins (both FrieslandCampina and retailers'). This explains why the subject of this project is important in a business context, where it is crucial to know price-driven changes and impacts.

For our scope, in reality every retailer responds to a change in price differently. Generally, retailers that are big in terms of sales and market share (for example Albert Heijn and Jumbo in The Netherlands), tend to be the first ones among all retailers to change their price in response to a change in the PTR and the remaining follow. It may also occur that the margin made by the retailers are much larger than that made by FrieslandCampina for a given product. In this case, the business goal may be to increase profitability. Here, increasing the price of the product to the retailer could be a preferred tactic. In order to undertake pricing decisions by understanding the underlying parameter space, a Visual Analysis tool in the form of a simulator that is capable of answering user questions through visualization techniques, could come handy.

Chapter 4

Problem Description and Analysis

This chapter describes the problems of concern that we aim to solve through our work. The first step toward understanding the parameters to be modeled for development of a visual analysis tool is explained in this chapter.

4.1 The problem

In the previous chapter, chapter 3, we saw how to classify a product as price elastic and inelastic; this can be done by computing the relation between proportion of increase in price and the consequent proportion of increase or decrease in the volume demanded (refer to section 3.5). With historical data of the product's volume sales at corresponding prices, a statistical framework that computes the price elasticity of products can be modelled. In case of FrieslandCampina, such a model should be able to incorporate price elasticity of the products at a retailer level because, as discussed before, each retailer would react to a hike or fall in price of a product in a different way. The problems can now be stated and discussed in the following way:

1. Which are the variables that contribute to a consequent change in Revenue KPI's when the price of a product or product-cluster is changed?

   This refers to the need of modelling price elasticity for each of the product-clusters to observe a change in their Sales Volume in response to price change. At the same time, it is necessary to know what happens to all the other variables dependent on Sales Volume in order to compute higher-level variables such as Margins. Hence, to answer question 1, mapping inter dependencies between the variables that contribute to financial KPI's is crucial. This will be discussed in the subsequent subsection 4.2.

2. How does price change impact the volume sold to retailers as a function of their respective price elasticities?

   This aspect aims at the analysis of the change in Sales Volume and Sales Value driven by change in price, as the change in these two metrics is most likely proportional.

3. What happens with the (total) FrieslandCampina margin when the price of a product-cluster is changed?

   This aspect aims at the analysis of the increase or decrease in the total FrieslandCampina margin (across all products and retailers) as a consequence of change in price. In some cases the analysis might demand just an overview of what happens with the total margins in response to price change.

4. What is FrieslandCampin's margin compared to Retailers' margins for a given product-cluster?

   It is obvious that that there will be a considerable revenue margin that the Retailers keep for themselves when they sell the products bought, to consumers. After a change in prices of products, if the consumer sales is still by far as expected, this means that the consumers are still willing to buy the products. Implicitly, this allows potential higher FrieslandCampina margins. By comparing FrieslandCampina and retailer margins, the distribution and share of absolute margins can be deduced for sketching a pricing approach under the hood of revenue growth management objectives.

   Additionally, if the end-goal is to obtain higher revenue or total margin for instance, then the first step would be to initiate and engage in an in-depth analysis of a category or brand influence on the revenue and margins. This means that if the change in price of a particular product-cluster pertaining to its corresponding category or brand is sufficient to fuel a noticeable increase in the revenue and/or margins, then the purpose is met. In questions 3 and 4, the contribution of individual retailers on the change in KPI's also becomes an interesting observation to make vital decisions for undertaking tactics.

5. What would the impact on FrieslandCampina's sales volume and margins be when their competitors in the same product segment change their price to consumers?

   The competitor's price for a product belonging to the same category or segment is an equally important indicator of gaining a realistic overview of pricing strategies and tactics. Imagine a situation where FrieslandCampina increased the price of a product, such as yoghurt by 1%. Now, if one their main competitors in the yoghurt sales segment decreases their price in response, then it could be the case that this has an influence on the sales of the former's yoghurt in the market. Therefore competitor analysis must always be accounted for in strategizing tactics. This is why this question adds a high value to our analysis.

6. Which kind of techniques would allow users to undertake a comparative analysis of different price scenarios?

   While carrying out such exploratory analysis, it becomes important to create different scenarios and further compare them to reach conclusions from the starting point of the questions in mind. Therefore, the problem stated here aims at finding methods which can enable comparative analyses of different price scenarios. The most important question is however, the chosen technique should make the analysis process meaningful, efficient and easy to grasp. To be precise, how well can visualization techniques help answer all underlying questions and enhance the take-away findings from the user's analysis journey.

7. Finally, how can such analyses be made more intuitive and insightful than the traditional means of using Microsoft Excel or business intelligence tools, for instance, to carry out data-driven investigations?

   It is possible to make computations and calculations using traditional means such as digital worksheets. However, the process of determining pricing tactics requires creation of a story that describes a specific scenario. Through many such stories and scenarios, the analysis process should allow users to understand the parameter space better for making more informed business decisions. As stated in this problem statement, how interactive visualization techniques can supersede other methods by taking user tasks and actions into consideration will be discussed later in this report.

4.2 Margin computational model

Prior to developing the visual analysis tool, the first step in the scope of this project was to reproduce a computational framework that models the price-dependent variables. This is illustrated in the figures 4.1 and 4.2, and aims at answering question 1 of our problem description.

In a bottom-up approach, these figures explain the Margin computations, both FrieslandCampina and Retailer Margins, based on the variables seen as attributes in the company's data sources. We call this a bottom-up approach because using the variables available at a granular level, we can compute and derive higher-level variables such as Margin. This also means that the FrieslandCampina and retailer margins are not directly available in the database but need to be computed. The retail sales information and data is obtained by FrieslandCampina from the company called Nielsen. They are a leading global information and measurement company who deliver critical media and marketing information, analytics and industry expertise about what consumers watch (Television, mobile and online) and what consumers buy on a local and global basis[14]. All the internal sales information is stored in the Internal Sales database of FrieslandCampina. These are the two databases from where all the concerned variables can be retrieved for margin computations.

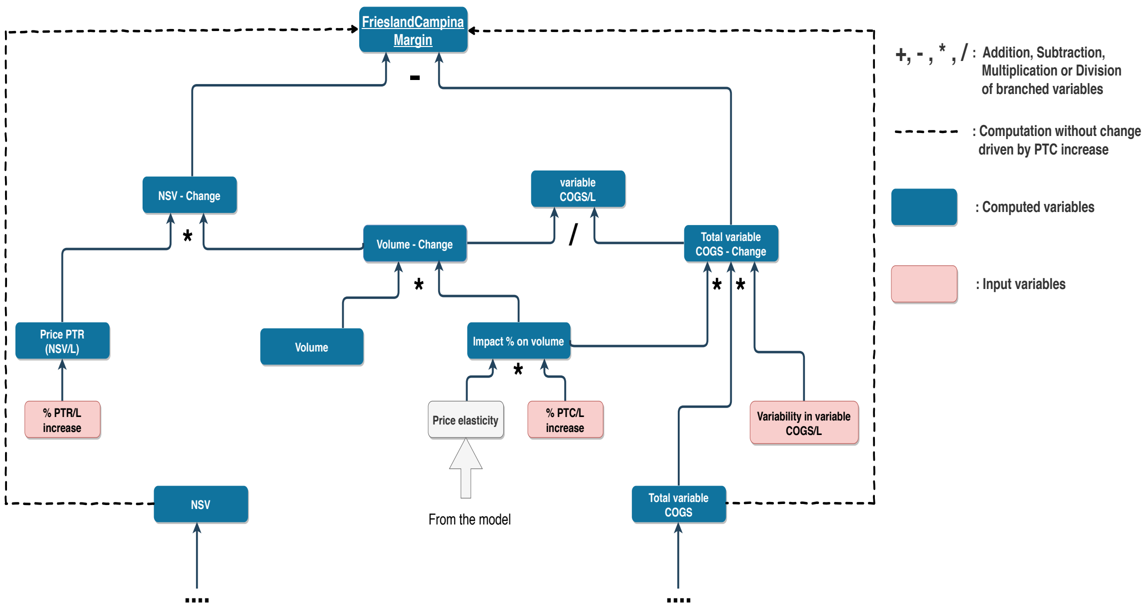


Figure 4.1: FrieslandCampina Margin computation in a bottom-up approach from a set of dependent variables

In Figure 4.1, a computational model has been created to focus on the higher-level variables required to compute FrieslandCampina margins in a bottom-up approach. By subtracting all the rebates, discounts and taxes from the Gross Sales, the Net Sales Value NSV can be computed. Similarly, the Total Cost of Goods Sold can be accounted for by summing up all the production and distribution and selling costs as shown. All these data attributes are derived from the Internal Sales database. For a full view of this model, refer to the Appendix B.

Climbing further up in the hierarchical model, the price of products are now introduced and can be modelled. Like illustrated earlier, there are two categories of prices, PTR and PTC and these are the inputs to such a model where impact of change in price is to be computed. Once the price elasticity values are obtained from a backend statistical model (shown as the grey input block in the diagram), this can be multiplied with the percentage increase or decrease in PTC per liter for computing the impact percentage on the Sales Volume. The new Volume (Volume change in

figure) will be the base Volume(in kilograms or liters) multiplied by the impact percentage. This portion has been explained in detail in section 3.5. For this project, this Price elasticity model is a black box input to this model as the details into it are not known or relevant for this project. It can thus be seen as input model which computes the price elasticity of products per Retailer.
On the other hand, a percentage increase in our second price parameter PTR per liter (shown on the left in the figure), gives a new rate of NSV per liter. The change in Total NSV as a result of this price change can now be computed by multiplying the per liter NSV by the Volume change.

On the right hand side of the hierarchical flowchart, we see how the Total COGS is impacted by a volume change as well as the parameter termed as "variability in variable COGS/l. The latter variable is a measure of a concept which suggests that as the volume demanded increases, the costs per liter would decrease. This is a measure which determines the variability percentage in the total COGS when the price is changed. However, for the scope of this project, this parameter will not be used in the model. It has been included in this map of dependent variables because this behavior is seen in reality and must be accounted for. Thus, we can consider the variability in the Totals COGS to be constant and say that the change in total COGS due to a change in price can be measured by multiplying the impact percentage on volume with the base value of Total COGS and this constant.

We now reach the apex of this variable hierarchy and can compute the intended FrieslandCampina margins by subtracting from the Net Sales, the total costs as shown in this map.

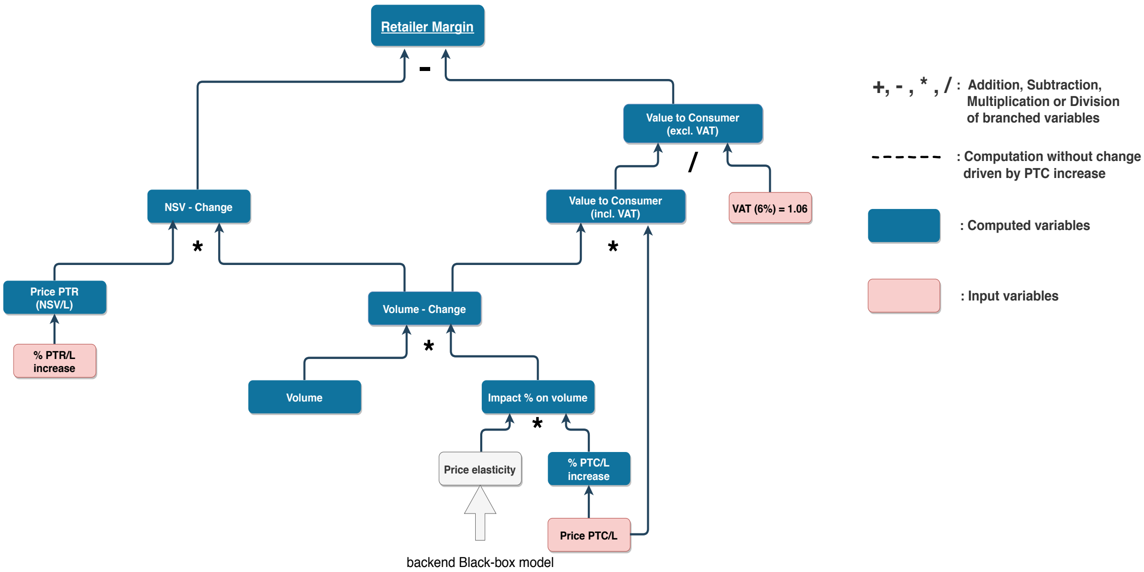


Figure 4.2: Retailer Margin computation in a bottom-up approach from a set of dependent variables

With the retail sales data, it is also possible to compute Retailer Margins, as illustrated in figure 4.2. This model is quite similar to the FrieslandCampina margin model with the same inputs to the model except that here, we incorporate the Value to Consumer as the revenue for a given retailer. The PTC per liter is the rate of the product per liter at which retailers sell the products in the market. On multiplying their per liter price to consumer with the total volume intended to be sold, the Value to Consumer which is the revenue for these retailers can be computed. This measure is also termed as the Sales Value, as seen previously in the report. However, this value includes the Value Added Tax (VAT) on the sales of the product which is thereby a cost which the retailers bear that must be subtracted from this quantity. Depending on each country, the VAT percentage is different. In this model, we have shown a 6% VAT that applies for retail sales on products sold in The Netherlands.

Finally, from the revenue earned by the retailer (Value to Consumer), the Net Sales can be subtracted to account for the margin made by the retailer. This gives us the desired Retailer Margin which obviously, is retailer specific. At this point in the report, we have a clear idea of how to carry out the computations to achieve the project's objective.

Chapter 5

Price Simulator

In this chapter, we build a foundation for our visual analysis tool, that we name as the Price Simulator. We show how the users would interact with the tool through an user-interface architecture. Details about the data available are also presented in this chapter. Lastly, the composed set of user tasks to guide the design and development process, are discussed.

5.1 User-Interface interaction

Visualization becomes the medium of an analytical process where the humans and machines integrate to cooperate using their respective unique capabilities. While designing an interface, in order to enrich it with interactive functionalities, the user has to be the ultimate authority in leading the direction of analysis alongside his or her specific tasks[10]. To support this, the system must provide the user with sufficient and effective means of interaction. These principles are important considerations for this project as we want the design of the visualization tool discussed in this document to be user-centric. For understanding the human-interface architecture, figure 5.1 shows a pictorial representation of the flow of the interaction process.

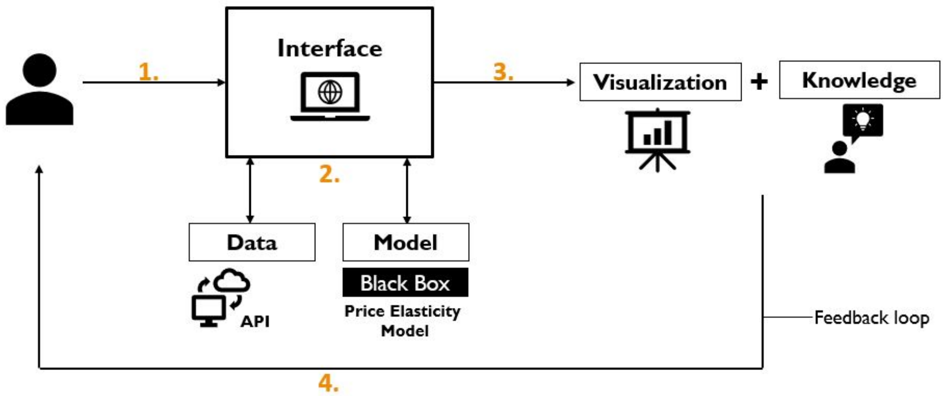


Figure 5.1: Flow of user-interface interaction. Step 1: user provides some input to the interface. Step 2: the interface fetches data and values from the Application Programming Interface (API) and backend price elasticity model, respectively. Step 3: the interface produces visual output for the user to gain insights from. Step 4: the user refines his analysis to meet his objectives as a feedback loop.

The user interacts with our interface by providing some inputs to it. The interface requests data from an API and inputs price elasticities from the so-called "Black Box Model". The reason why we call this a Black Box is because how it computes the price elasticities is not known and therefore deemed unnecessary for the visualization scope of this assignment. So, two external input components, other than the user's activity as an input to the interface, are the data pipeline and the black-box price elasticity model. The interface is an interactive web application that will be rendered on the user's web browser to provide a simulation environment with functionalities to be discussed further in this report. After the system fetches the data attributes and price elasticity values corresponding to the user's input parameters, it produces output in form of visualization graphs and charts. A considerable amount of time was also dedicated for understanding the type of questions that the visualization output should support, for the users to obtain knowledge in return. Only then the purpose of developing a user-specific platform is met. However, to reach their objectives, the users take an iterative approach for refining their inputs and adapting the interface to produce desired output. This is essentially the feedback loop that binds every new iteration of analysis with the previous one for attaining knowledge and discovering new findings.

Before we explain the hypothesis and the user tasks in detail, let us try to briefly state the basic capabilities of an interface that have been taken into consideration for designing and developing it. This is shown in figure 5.2.

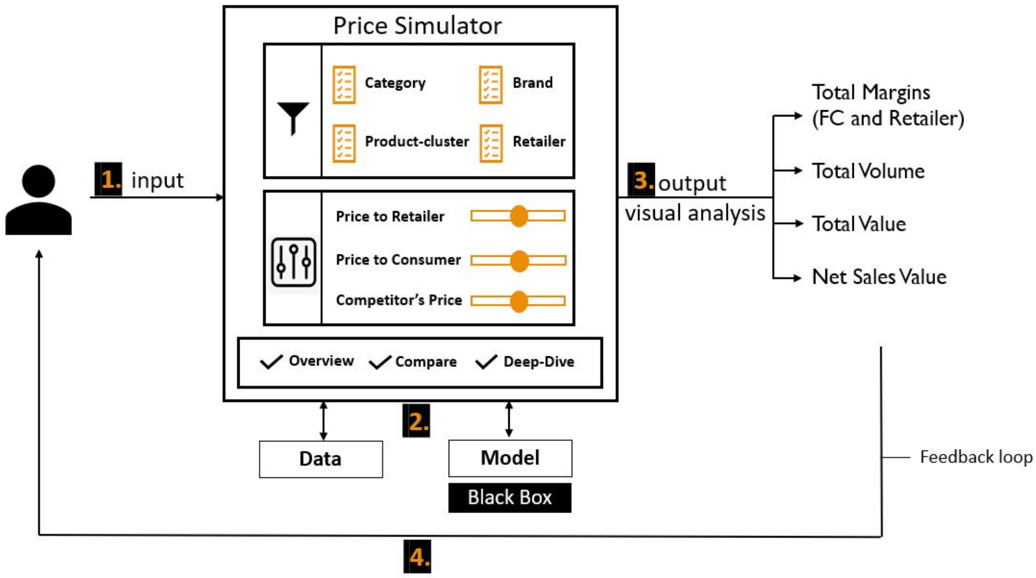


Figure 5.2: The interface is called the Price Simulator. The basic user interaction would be introduced through tasks such as filtering on the data attributes and changing price values. The three higher-level user tasks can be categorized as Overview, Compare and Deep-Dive analysis.

If we try to zoom into the "Interface" component of the illustration shown in figure 5.1, we see that it requires the following functionalities:

1. The user should be able to filter on various data attributes such as Category, Brand, Product-cluster and Retailer.

2. Upon filtering the data for a targeted analysis, the interface should support a functionality that enables the user to tweak price parameters. As for the problem description, the user would like to change the PTR and PTC per Category/Brand/Product-cluster/Retailer. In addition, the pricing data for competitor brands is also available for gathering realistic observations. Hence, the interface should also accommodate a feature that lets the user change competitor prices to analyze the impact on the variables of interest.

3. The higher-level tasks or actions that the users would engage in with the help of this interface are:
   (a) gaining an overview of all the Revenue KPI's involved at once; the current scenario (illustrating the present pricing scheme) and forecasted scenario (quantifying the total change or deviation from the current scenario if prices are changed),
   (b) comparing several price scenarios that have been simulated by the user, and
   (c) gaining a deep-dive analysis of each of the KPI's for detailed analysis of the influence of price change.

It is also important to state who the users of this system are and introduce the framework to be used for the development of the Price Simulator. Let us state these aspects one by one.

1. Who are the users?
   Since this project is a proof-of-concept for supporting pricing tactics, the end-users are the Digital Media and Insights Team itself (the team under which the project is being developed). More specifically, our users are data-savvy individuals, who deal with activities such as data cleaning and reporting on a daily basis. While they are experienced in creating visualizations of complexity ranging from low to high, the end-users are capable of using data visualization as a means for undertaking daily business-oriented tasks. Besides, they are typically interested in observing trends in the data and using insights from these visuals to carry out more mature analyses.

2. What is the chosen framework for the development of the proposed Price Simulator?
   The development platform chosen for this project is **Dash** framework of Python by **plotly** for building an analytical web application. In the system described in figure 5.1 and 5.2, we saw the need of interaction in our interface. The fact that Dash is developed on top of javascript libraries such as *Plotly.js*, *React.js* and *D3.js*, makes it a suitable candidate platform for web development. Components such as sliders and dropdown menus that help in filtering data, and graphing components that help in creating visual charts can directly be tied to a python script without the need of javascript.

3. Which type of a device do the users intend to view and operate the web application on?
   The assumed device for the usability of the web application is a desktop computer. The design choices discussed further in chapter 6 of the report, adhere to this assumption.

Before proceeding with user tasks, let us first describe the data available for implementation of the Price Simulator.

5.2 Data description

As we have seen in figure 5.1, the data to be used for implementation is fetched through an Application Programming Interface (API), which publishes the data in JSON (JavaScript Object Notation) format. We try to represent the data attributes available with the help of figure 5.3. For each product-cluster, the *Category*, *Segment*, *Brand* and *Parameters* information is available. Under the *Parameters* data attribute, the retailer names for the product-cluster's brand and the name of similar competitor brands are available.

Further, let us explore how the data and price elasticity model values are fetched to the interface upon user interaction, through figure 5.4. Firstly, when the user inputs the choice of "product-cluster" via the interface (Step 1), the corresponding "Parameter" attributes are fetched onto the interface through the API (Step 2). This data attribute dependency is seen in figure 5.3. With this, as seen in Step 3 of the figure, the interface is now populated with price slider components per retailer and competitor brand name. When the user changes the price percentage to a value

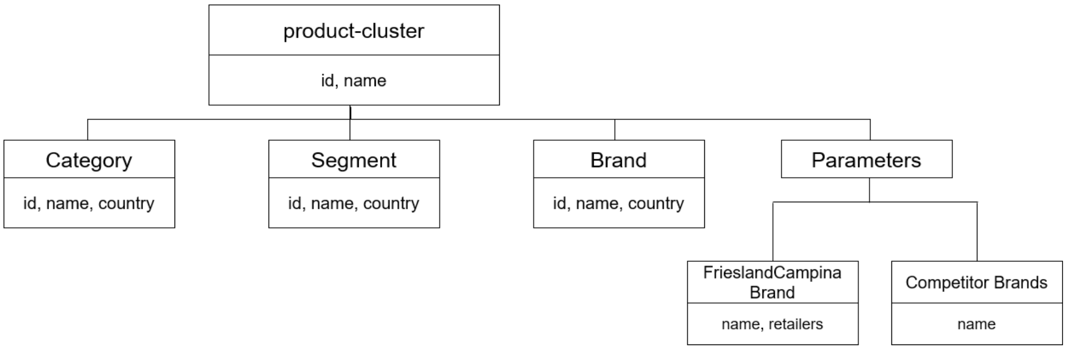


Figure 5.3: Data and attributes available in a JSON format via API call. For each product-cluster, the category, segment, brand and retailer information is available.

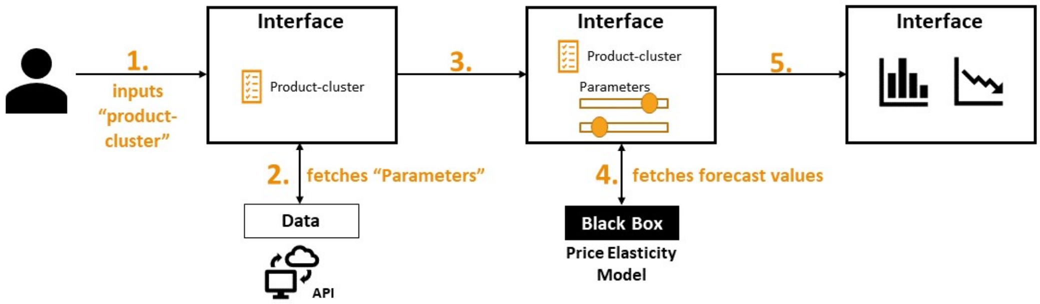


Figure 5.4: Illustration of how the data and price elasticity values are fetched from onto the interface. Step 1: user inputs the product-cluster value via the interface. Step 2: based on the chosen product-cluster, an API call fetches the corresponding "Parameters" attribute. Step 3: based on the parameter attributes, the price sliders per retailer and competitor brand are displayed on the interface. Step 4: based on the price settings, the black box model computes the price elasticity of the product and retailers, and returns the Sales Volume and Sales Value forecast values. Step 5: the interface of the Price Simulator can now support visualization graphs based on the data.

between -5% and 5%, the backend black box price elasticity model computes and returns the forecasted values for Sales Volume and Sales Value, as a function of the product's and retailer's price elasticity value. This is seen in Step 4 of the figure. We do not know what this black box model looks like and how it performs computations. The price elasticity model has only been implemented for FrieslandCampina's brand *Landliebe*, in the segments of *Ready To Eat (RTE)* and *Desserts*. Based on the forecast values fetched, we can create visualizations for data representation on the interface, as seen in Step 5 of the figure.

Based on the data set available, the next step is to determine user tasks for creation of suitable visualizations for the interface. This is discussed in the following section.

5.3 User tasks and hypothesis

The tasks of the users are an important constraint for a visualization designer to account for questions that the users wish to answer through the visual analysis tool. In context of our use case, the analysis to be carried out by the users is based on the following hypothesis:

- **H1**: When the price of a product-cluster is increased per retailer(s), while the profitability might see an increase, the sales volume will most likely decrease.

The higher-level objective of developing this visual simulator aims at making the validity or invalidity of this hypothesis evident through visual representation of data.

Apart from validating the hypothesis through visualization methods, our objective under the project's scope is to achieve user's tasks. These are understood as the activities to be carried out interactively on a visual data representation for specific reasons. Investigating these tasks helps in improving the design and evaluating it in the end. The combination of **Data** + **Task** asks which **Visualization** is best suited to pursue a given task on the user-specified data input. This aspect caters to a visualization design. In figure 5.2, we tried to summarize some user tasks on an abstract level. We now list and discuss eight distinct user tasks that were determined for designing the Price Simulator.

The authors Schulz et al.[18] present a design space for visualization tasks. They describe a design space as a combination of a finite number of design dimensions. Each of these capture one particular design decision which as a whole, fully specify the design space. The approach discussed in their work has been used for defining the five dimensions related to the "5W's" of WHY, WHAT, WHERE, WHO and WHEN, as well as the HOW. The WHEN and WHO are not inherent fundamental properties of an individual task itself, and rather aim to relate a task to its context. We therefore do not include these two aspects in our design space. First, let us begin by describing what the dimensions WHY, HOW, WHAT and WHERE represent.

1. **WHY** is a task pursued? This defines the **goal** or the intent with which the visualization task is being pursued.
2. **HOW** is a task carried out? This defines the **means** of reaching the goal.
3. **WHAT** does a task seek? This defines the properties or **characteristics** of the data that the task tries to disclose.
4. **WHERE** in the data does the task operate? This specifies the **target** and the **cardinality** (scope or range) of data entities within the target.

It is a good practice to gather or formulate user tasks and answer a set of such questions as these motivate our design choices. Moreover, by answering these questions, the designer ensures that correct and efficient visualization schemes are used for allowing the user to meet his or her goals. By understanding the scope of the problem and interrogating the users, the following tasks were identified. Each individual task has been represented as a 5-tuple *(goal, means, characteristics, target, cardinality)*. More explanation about the terminologies and notations used can be found in Appendix C.

T1 : gain overview of all the KPI's involved

- (exploratory, summarize, trend|data values, attrib(Sales Volume)|attrib(Sales Value)|attrib(Net Sales Value)|attrib(Margins), all)

  This means that the user is *summarizing trends* and *data values* among *all* available *Volume, Value, Net Sales, Margin attribute values.* This task is *exploratory* as the user does not yet know in particular about the absence, presence or the nature of the trend that he or she will observe in the data.

T2 : filter data on product Category, Brand and Product-cluster

- (confirmatory, browse|filter, data values, attrib(*Category)|attrib(*Brand)|attrib(*Product-cluster), multiple)

  This means that the user is *browsing* and *filtering* through *multiple Category, Brand and Product-cluster attribute values.* This task is *confirmatory* as the user makes an informed choice for a targeted analysis.

T3 : change Price per Retailer and/or Competitor brand

- (exploratory|confirmatory, reorganize, data values, attrib(Price)|attrib(Brand), single |multiple)

  This means that the user is *reorganizing* (adjusting) the *data values* of *single* or *multiple* instances of *Price* attribute that is available per *Brand* attribute. Depending on whether or not the user knows which price parameter to tweak for meeting his targeted goal, this task could be either *exploratory* or *confirmatory*.

T4 : select variables for scenario comparison

- (exploratory|confirmatory, lookup, data objects, attrib(Sales Volume)|attrib(Sales Value)| attrib(Net Sales Value)|attrib(Margins), all)

  This means that the user is *looking up* the *data objects* among *all* available *Sales Volume, Sales Value, Net Sales Value, Margin attribute values* for selection. To initiate this task, if the user has a goal in mind for comparison, this task is *confirmatory* and otherwise, *exploratory*. This task can be considered as a prerequisite for user task **T7**.

T5 : analyze trends in the KPI variables

- (exploratory, search, trend, attrib(*)|attrib(time), all)

  This means that the user is *searching* for *trends* among *all* available attribute values of *all* the KPI variables (denoted by *). Clearly, this is an *exploratory* analysis.

T6 : analyze the change in KPI values driven by price change (forecast values), with respect to the baseline value

- (exploratory, relate|compare, trend|distribution|data values, attrib(forecast) |attrib(baseline), all)

  This means that the user wants to *relate* (for similarities) or *compare* (for differences and deviations) among *all* temporal *forecast* values with respect to the *baseline* values for each of the KPI's. This task can be categorized as *exploratory*.

T7 : compare different price scenarios

- (exploratory, compare|relate, features, attrib(KPI)|attrib(Price), all)

  This means that the user wants to *compare* or *relate* several desired *features* from *all* available attribute values of the desired *KPI's* against the *Price* value chosen for simulation. The nature of such a task would also be *exploratory* as the outcome is not yet known and in fact, this is what the user would like to learn.

T8 : derive each retailer's contribution to the change in Sales Volume and Margins when its price is changed.

- (exploratory, derive|compare, data attribute, attrib(retailers), all)

  This means that the user wants to *derive* the contribution as a *data attribute* among *all* the *retailers* and *compare* the derived contribution values. This is an *exploratory* analysis where the user attempts to find out additional information that is not directly present in the data.

Together these tasks form a set of typical user actions for engaging in exploratory analysis of observing the impact of price change on KPI's and comparing different price scenarios.

Chapter 6

Design

This chapter discusses in detail, the design principles and choices made to develop the interface for the visual simulator. The design decisions made for each individual user task or in some cases, a combination of user tasks (laid down in section 5.3 of chapter 5) are elucidated.

6.1 Design principles

Design principles can be defined as the chief choices and decisions made in order to develop a user interface that supports visuals for data representation. For designing an effective and efficient user application, defining the layout of its interface, choosing suitable visualization techniques, and several other aesthetic facets need to be explored. The design aspects of application development is often iterative and involves inputs from end-users. In our context, since we have a set of predefined user tasks, some design decisions were user-determined while others were for the designer's to make. The choices made and discussed further in this chapter aim at providing our potential users with a tool that is user friendly with visual schemes that are relevant to the user tasks, while offering interactive functionalities. Since our visual analysis tool is to be designed in the form of a simulator, some components have been chosen to achieve that kind of look and feel.

In the beginning of chapter 5, we claimed that the user tasks, on an abstract level, should allow the users of the tool to *gain an overview* of KPI's, engage in *deep-dive analysis* and *compare* different price scenarios when simulated (refer to figure 5.2). In order to make the design simpler, the design space could be divided into two different views by categorizing the three tasks as shown in the figure 6.1.

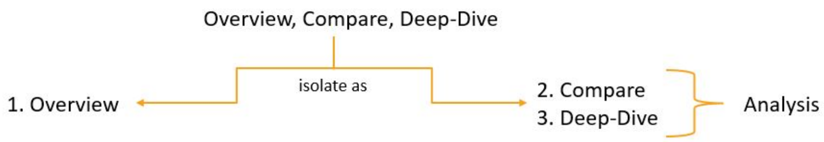


Figure 6.1: Categorizing the 3 higher-level tasks as *Overview* and *Analysis*.

6.2 Split views for *Overview* and *Analysis*

The first view of the interface, could be designed exclusively for gaining an *Overview* of all the KPI's; the total Sales Volume, total Sales Value, total Net Sales Value and total Margins of FrieslandCampina and Retailers at the current (unchanged) price, across the whole year (aggregate over August 2019 to August 2020). The second view could be designed for carrying out extensive *Analysis*; deep-dive into how price change impacts the KPI's and comparing trends, behaviors at different price settings. With this idea, the web application interface has been divided into two web pages for allowing the user to navigate to desired location for carrying out related visualization tasks. Figure 6.2 shows the design mock-up of the Price Simulator, with the FrieslandCampina logo on the top-right corner, and the navigation menu for the two pages at the top-left position of the view.

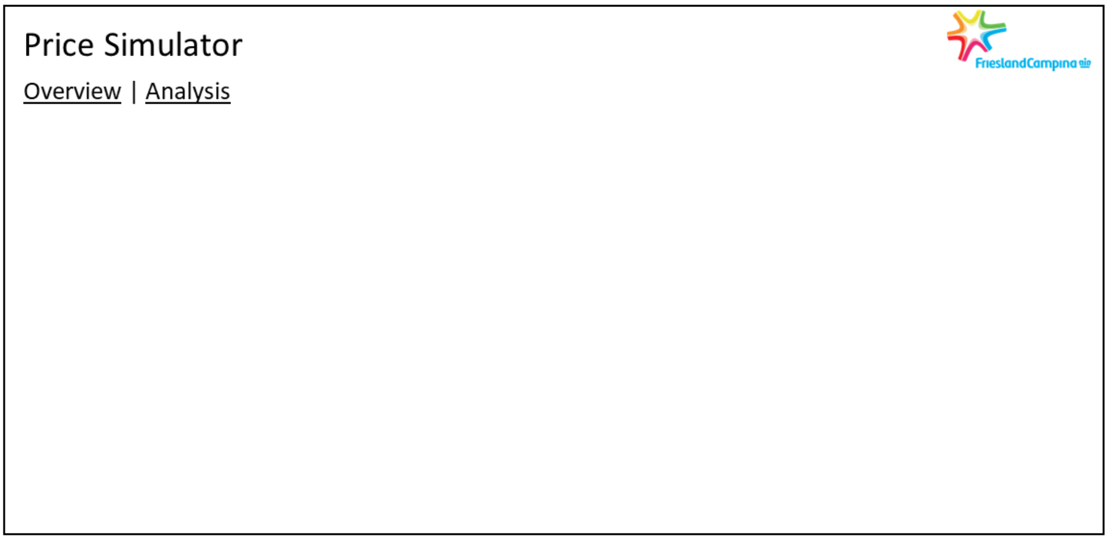


Figure 6.2: Hyperlinks to navigate between Overview and Analysis page of the Price Simulator.

Let us take this mock-up as a foundation of our user interface and build it up further, task-by-task, for a better understanding of the chosen design concepts.

6.3 View for the overview of KPI's

Design for user task T1: The aim of this user task is to gain an overview of the KPI's involved. For high level strategic users who are mainly interested in gaining an abstract view of the KPI's to answer questions like, what is the total FrieslandCampina margin, an overview is the best way to summarize the key aspects. While this could be done in many ways, we propose a simple design where the four KPI's are displayed as four distinct blocks in the design space. A rough sketch of the resulting layout of this design decision, is shown in figure 6.3.

For each variable, the Baseline value gives the aggregate over the entire time period of one year, in a numeric format. These four sub-plots could be further enhanced by taking these numeric totals, and superimposing them on top of trend charts, where the latter represent the trends of these variables over time. In this way, the numeric totals would be highlighted and the trend lines corresponding to each KPI would act as a watermark.

To provide an abstract view of all four variables, the Overview page serves the purpose of a landing page for the application.

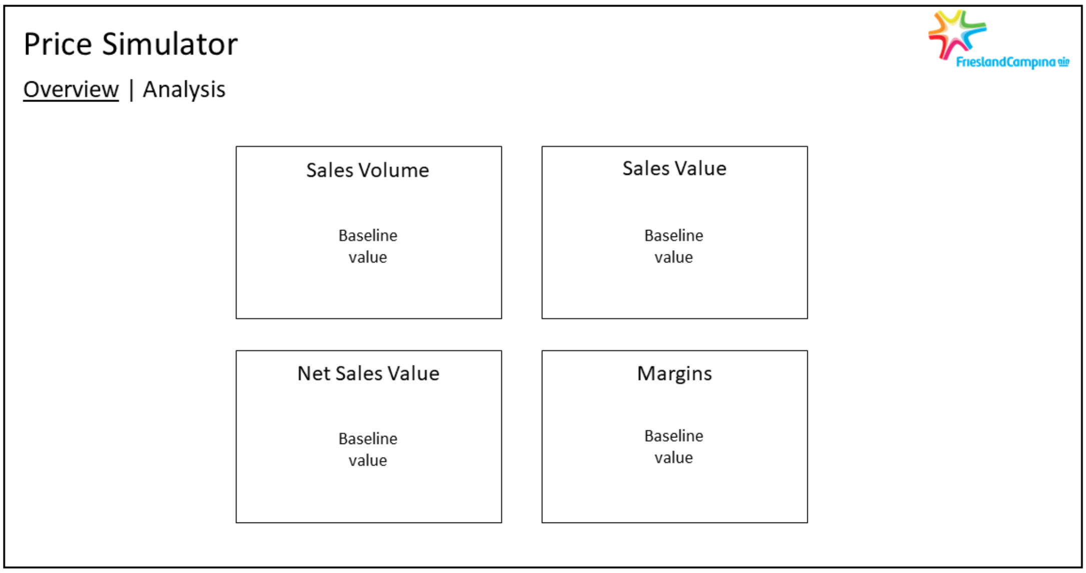


Figure 6.3: Displaying the four KPI's as four blocks on the interface's Overview page. The *Baseline value* depicts the sum of the corresponding variable over the time period in a numeric format. i.e. Total Sales Volume, Sales Value, Net Sales Value and Margins of FrieslandCampina and Retailers over the entire year, at the current base price.

6.4 View for filters and price settings

Design for user tasks T2 and T3: As shown in figure 5.2 in chapter 5, this user requirement involves filtering on Category of products, the Brands available within that product category, and the product-cluster under that Brand. Further, as we saw in figures 4.1 and 4.2 in chapter 4, there would be two main price inputs for the visual simulator, PTR and PTC per liter, and the user would like to change these two price percentages per Retailer for the chosen model of product-cluster. Additionally, it would also be interesting to visualize scenarios where competitor brands change their price percentage (to consumers) in a certain way. This leaves us with three price parameters that must be accommodated in our design for the user to tweak and analyze.

Among all the eight user tasks that we have seen in section 5.3 in chapter 5, filtering can be recognized as a recurring task that helps the users refine their search. For ease of use, we divide the design space into two sections; the left section for pursuing filtering tasks and the right for visualization tasks. This has been illustrated in the mock-up shown in figure 6.4 for the Overview page. The same design principle can be applied for the Analysis page. The design choices we propose for the filter and price settings are discussed below:

1. **Drop-down menus for filtering**: Single select drop-down menus can be used for filtering data attributes such as Category, Brand and Model of product-clusters. As one of the standard user input components, drop-down menus would help users choose an item from a given list. It is also flexible in containing any number of options, without having to adjust the design.

2. **Sliders for price settings**: Sliders help users change the value of an attribute in both upward and downward direction by the means of dragging the slider pointer. All sliders have starting and ending point which show the broadness of value user can input. In our context, this would allow the user to change the price percentage between a range of -5% to 5%.

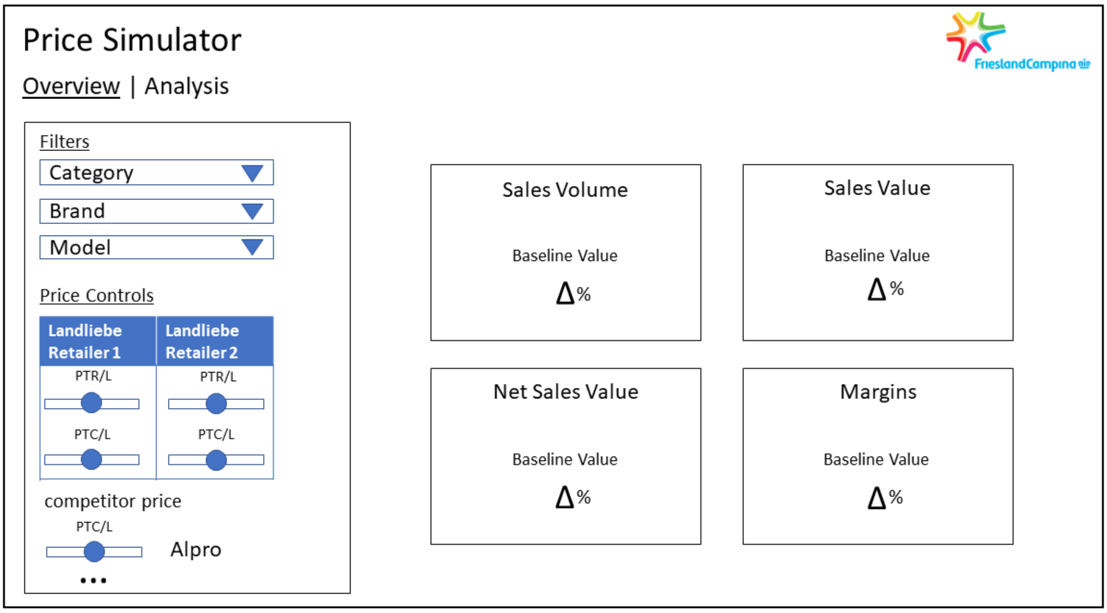


Figure 6.4: The Filter and Price Control parameters together have been fixed on the left side of the display and visual plots to the right. The filter parameters can be selected using drop-down menu. For FrieslandCampina's brand (Landliebe), sliders for price PTR and PTC can be placed within a data table. For each competitor brand (such as Alpro), a price slider for PTC can be provided.

- **Retailer Sliders**: Per retailer, our interface should comprise of two sliders, one for changing price PTR of the retailer, and another for changing price PTC of the retailer.
- **Competitor sliders**: The design should be extended to accommodate price sliders for competitor brands, shown as Alpro in figure 6.4.

3. **Three dimensional Data Table**: The choice of using a three dimensional data table for changing the price per retailer, would provide a comprehensible view to the users. In the three dimensional data table, the table row would represent the product-cluster selected in the Model drop-down, the table columns would be the retailer names pertaining to the Brand, and the cell values would represent the price sliders, two for the two price parameters we have. In this way, the data table can structure and organize a fixed number of sliders for a variable number of retailers per product-cluster.

Now that we can change the filter and price settings, we will now be able to predict scenarios to know how the baseline values of the KPI's change when the price is changed. In order to visualize the *change* with respect to the baseline, we can add a $\% \ \Delta baseline$ metric the KPI visualizations on the right sections of the view in figure 6.4, to convey the overall percentage uplift or downfall from the baseline. Further in this report, we develop the design to be adopted and visualizations to be used for the right section of the Analysis page in relation with the subsequent user tasks.

6.5 View for selection of variables for comparison

Design for user task T4: The aim of this visualization task is to solve the problem of comparing two variables. The user should be able to choose the two variables to compare from the four available variables. Several means of enabling such comparison have been discussed in Appendix D, along with their pros and cons. Out of the many possibilities, the chosen approach for tackling this problem, is explained as follows.

In figure 6.5, we propose the use of juxtaposition technique of visual comparison [6]. By placing two distinct variables next to each other, it possible to compare them efficiently. We propose fixing one of the four variables on the left with an enlarged view. The user can further click on the small multiples stacked vertically on top of each other (located on the right end of the view), to expand and compare it against the first, fixed variable. The selected variable's view appears next to the fixed variable's view, on the location referred to as "Default view of second variable" in the figure. The two views share the same size and proportion to facilitate ease and efficiency in comparison. At the same time, the user can also gain an overview of the values and trends of the variables viewed as small multiples, for the same filter and price setting. This way, the design is flexible in allowing the user to choose the variables they wish to compare. Toggling between different views with the help of the proposed click and expand functionality, would help the user analyze all aspects of the simulated price scenario.

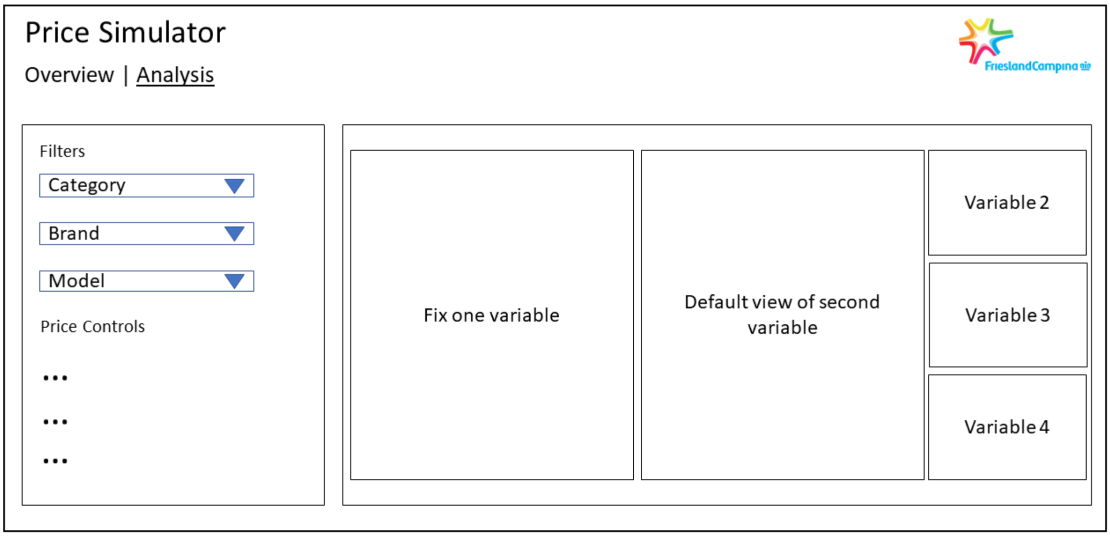


Figure 6.5: Fixing first variable. Setting a second default view. Small multiples on the right with "click to expand" interactivity.

Now that we have laid out basic functionalities and layout designs for our multi-page web application, we can proceed toward the next set of user tasks that focus more on answering visualization questions.

6.6 Visualization techniques

Design for user tasks T5 and T6: While the user task **T5** underlines the action of analyzing trends in the KPI variables, user task **T6** refers to the need of visualizing the change from their baseline values due to the influence of change in price. For the ideation of visualization methods capable of capturing these requirements from the data, the potential users were particularly interviewed about what is it that they would like to learn about the KPI's under the influence of change of price. The following responses and requirements were recorded:

R1 To visualize the overall trend of the KPI's over time for a given product-cluster.

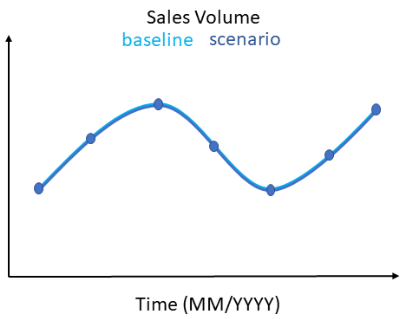
R2 To visualize the total percentage uplift or downfall of the forecasted scenario from the baseline as a reference, when price is changed.

R3 For ease in carrying out further computations using the visual simulator, the absolute difference between the baseline value of KPI and the forecasted value (as a result to change in price) at each time point should be a part of the visualization.

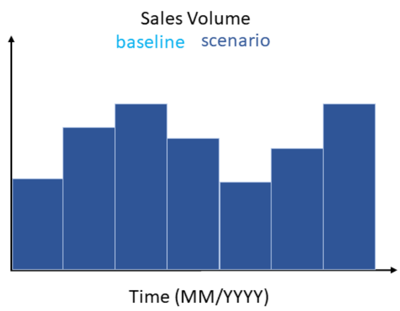
The process of choosing suitable visualization methods should first begin with the analysis of the type of data available and second, the type of data relationship to be conveyed through visual representation in our context. Since our data is a time sequence where we have the weekly baseline and forecast values of the KPI's over one year, it is most natural that our visualizations require taking time on the horizontal x-axis. This means that the data relationship that is important in our context is visualizing the change over time. For this, using time series visualization techniques can be meaningful. The most common time series visualization technique is projecting the time-dependent data as a line graph against a measure on the vertical y-axis such that the peaks and troughs can be spotted explicitly. In general, patterns such as seasonal behaviors, and trends such as gradual or sudden rise or fall in the measure values over time, can be communicated with this technique. These principles and features of time series visualization techniques could be used in favor of our user task under discussion.

We can aggregate our weekly data into monthly form to visualize trends and patterns over months of the entire year.

Let us examine some visualization methods that could be considered well suited for requirement **R1**. Figure 6.6 illustrates two possible data visualization methods to present trends over time,



(a) Line chart with smoothed peaks at time points to show the overall trend of baseline and scenario of Sales Volume.

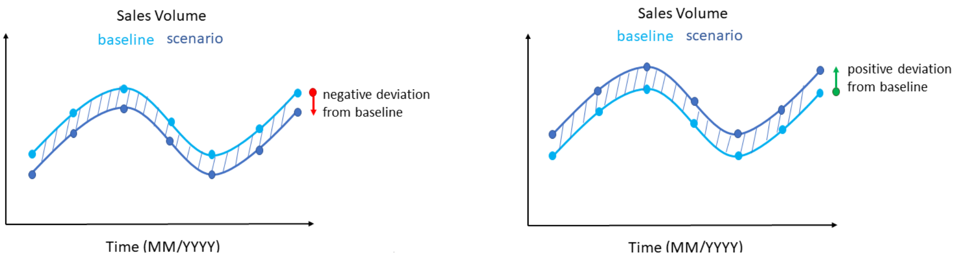


(b) Histogram to represent the trend with the height component quantifying the amount of Sales Volume.

Figure 6.6: Two simple possibilities of seeing trend of variables over time (Sales Volume variable taken as an example here). The legend colors could be used to represent the baseline and scenario traces distinctly.

simple line chart and a histogram. We propose using the superposition of baseline and scenario line traces. When the price is not changed by the user, the baseline and scenario traces superimpose exactly on top of each other to reflect no change in the baseline scenario, as seen in the figure. Further, we are interested in seeing if these two methods are capable and efficient in handling deviations of the scenario trace from the baseline, when the user tweaks the price settings. The data relationship to be conveyed here is the deviation of forecasted scenario from the baseline. Let us analyze how we could extend the use of these two methods to draw insights about deviations.

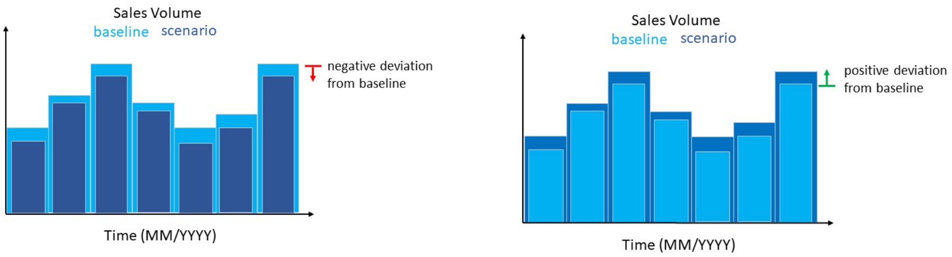
From the mock-ups shown in figure 6.7, we can see how line charts can be used to superimpose the two traces, namely baseline value and scenario's forecast value, in order to visualize the deviation of the latter from the former. By highlighting the area swept by the scenario trace on deviating from the baseline trace, the deviation of the scenario trend per time point can be made visually prominent. Apart from the overall trend, the change from the baseline can also be effectively communicated through such a representation. This visualization technique is popularly known as *Stream graph* which is efficient in visualizing changes in proportions over time[11].



(a) Negative deviation of the scenario trace from the baseline trace. The area between the two traces can be filled with a color to make the deviation explicit.

(b) Positive deviation of the scenario trace from the baseline trace. The area between the two traces can be filled with a color to make the deviation explicit.

Figure 6.7: Stream graphs of baseline and scenario value traces superimposed on the same graph. Positive and negative deviations of the scenario from the baseline can be analyzed with the help of color-fill feature between the two traces.



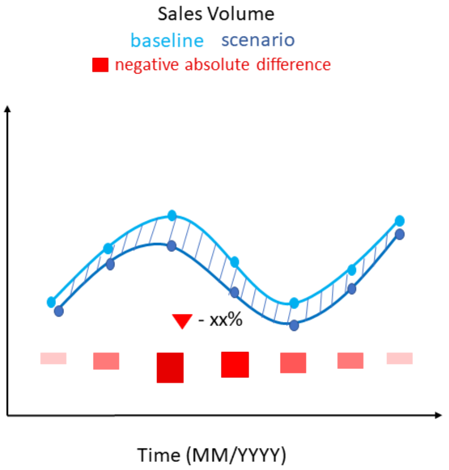
(a) Negative deviation of the scenario trace from the baseline trace using overlapping histograms.

(b) Positive deviation of the scenario trace from the baseline trace using overlapping histograms.

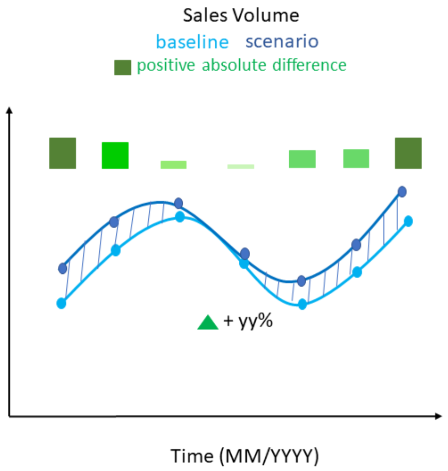
Figure 6.8: Overlapping technique to implement superimposition of baseline and scenario histograms can be used in this way. The colors and height comparison can help us draw insights about the deviation of the forecasted scenario from the baseline value.

In case of histograms, as shown in figure 6.8, the bars representing baseline and scenario can be overlapped for the superposition technique of visual comparison. In place of using the area property as used in the line graphs, this method uses the height component to show the difference between the two traces per time point. However, one downside of this method that may be realized later when we use it with actual data is that, for a small percentage change in price, the consequent change from the baseline value, i.e. the forecast scenario value, might not be large or significant enough for this visual method to depict explicitly. This is because, the height raised or lowered from the baseline value might not be visually apparent or noticeable. In contrast, our proposed Stream graphs would be able to handle smaller percentage changes from the baseline value better, with the fill-color feature. Besides, we cannot be sure that our web development framework will automatically bring the bar traces with a lower y-axis value to the front and push the bar traces with a higher value to the back, as we expect through figure 6.8. Hence, we do not consider histograms as a viable option for suiting this particular visual behavior that we expect would take place.

Let us now explore ways of meeting the requirements **R2** and **R3**, i.e. how can we extend our proposed visual graphs to represent the total percentage change over time, and the absolute differences between the baseline and scenario per time point. We propose a way to combine the three requirements through one graphical representation so that all the information is available at the same point in the design space, without clutter. This is illustrated in figure 6.9.



(a) Bar graph with y-axis range restricted to negative values can be used to display the negative absolute difference between the scenario and baseline values per month. The red colorscale gradient shows how large or small the absolute difference is.



(b) Bar graph with y-axis range restricted to non-negative values can be used to display the positive absolute difference between the scenario and baseline values per month. The green colorscale gradient shows how large or small the absolute difference is.

Figure 6.9: Shared x-axis graph with three y-axes. The Stream graph and bar graph combination shows the trend of Sales Volume over the shared time axis. If the price change causes a positive deviation from the baseline, the top y-axis is triggered and if the deviation from baseline is negative, the bottom y-axis is triggered. The bar graphs show the absolute difference between the scenario and baseline values per time point. The delta indicator used as an annotation in the center of the graph can be used to represent the total percentage increase or decrease over time.

In order to show the absolute difference between the scenario and baseline values per month, we propose using bar charts along with the Stream graph by dividing the y-axis scale into three components on a shared x-axis. To visualize the negative difference from the baseline value, the lower y-axis (component 1 of our y-axis) supporting only negative values of the (scenario - baseline) difference can be triggered, as shown in figure 6.9(a). The red color gradient can be useful to see at a glance, how large or small the difference from the baseline is at each time point depending on the color intensity. Similarly, in figure 6.9(b), we restrict the scale (component 2 of our y-axis) to only non-negative values. This means that when the (scenario - baseline) difference is positive, this bar graph can be triggered to show the positive differences per month. A green continuous color scale indicates the positive difference through the color intensity. The positioning of the two bar graphs in this way makes visual analysis effective by providing a sense of direction to the deviations from baseline. To indicate the total percentage increase or decrease across the whole year, the graph can be annotated with an indicator of a Δ symbol to denote this. The direction and color of the percentage Δ can bring out the underlying message as an overview.

Another alternative to using a combination of Stream graphs and bar graphs to achieve the three requirements being discussed would be to use Horizon graphs[15]. A horizon chart shows the behavior of a metric over time, keeping a baseline or horizon as a reference. It can support both positive and negative deviations from the baseline. The graph mirrors the negative values to the positive y-axis, where the positive and negative values can be differentiated using different color schemes. An example of this kind of a chart is illustrated in figure 6.10.

Until this point, we tried to discuss possible methods of visualizing Sales Volume. As seen in the computational model in chapter 3, the Sales Value is a measure of the Sales Volume multiplied

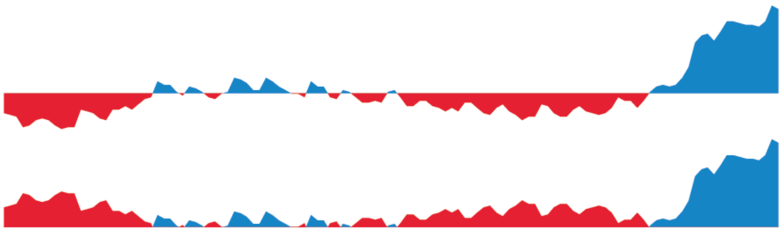
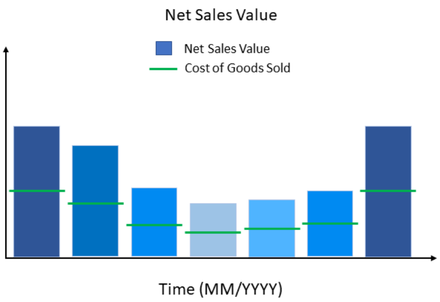
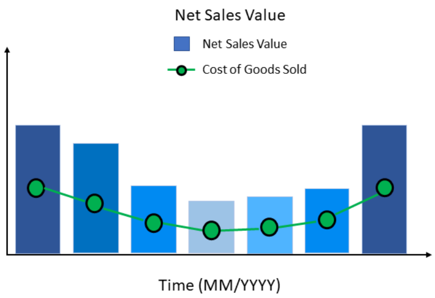


Figure 6.10: Horizon graphs of U.S. unemployment rate, 2000-2010[15]. Positive values indicate above average unemployment with the help of blue color; negative values indicate below average unemployment with the help of red color.

by the price. In this way, the variation in Sales Value will be proportional to that of Sales Volume as the former is a derivative of the latter. Thus, we could use the same visualization schemes for representing the Sales Value variable on our interface. Let us now try to sketch possible visual graphs for Net Sales Value and Margins.



(a) A Bar graph for Net Sales Value over time with a Scatter plot with horizontal line as data markers to represent the Total COGS per month for comparison.



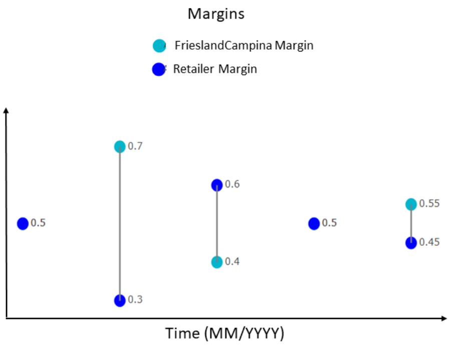
(b) A Bar graph for Net Sales Value over time with a Line+Scatter plot to represent the Total COGS per month for comparison.

Figure 6.11: Combination of Bar graphs with Scatter and/or Line graph for representing the Net Sales Value superimposed with another graph for Total Cost Of Goods Sold. This provides a way to compare the Revenue against Costs.

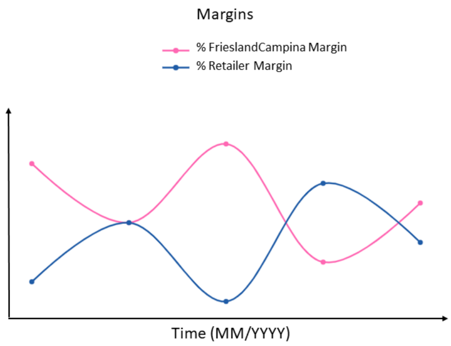
For visualizing the Net Sales Value, we wish to gain knowledge of the Revenue earned per month. To express a quantity where knowing the "how much" is important, we can choose a visual encoding technique that encodes the quantity with size or height as a dimension. A bar chart can be used for this purpose, where the height of bars can be used to encode and quantify revenue. However, the amount earned must always be measured against the amount spent, in a typical business context. We can therefore include costs in our visualization.

There could be two ways to visualize the total COGS against the Net Sales Value represented using graphs. One could be by showing the total costs per month as a reference or threshold. This is what we see in figure 6.11(a). The costs per month could be visualized as a scatter plot with its data marker being a horizontal line as reference. If the height of the bar is below this line, then it clear that a loss is borne in that particular month (as costs exceeded the revenue). In figure 6.11(b), we use a line graph with circular data markers for representing the total costs against revenue. In a lower resolution view, the use of this technique to express the overall trend against Net Sales Value could be effective. However one possible drawback could be the identification of the exact center location of the circular data markers at a glance, to gain an overview of costs per month. Thus, it may be better to use horizontal lines rather than circular data markers for encoding the costs per month.

Lastly, we propose two visual methods that are efficient in the analysis of our fourth KPI variable, the Margins of FrieslandCampina and Retailers. On computing the margin pool (refer to definition in section 3.4), we can also obtain the percentage of margins made by FrieslandCampina and across all Retailers, which together add up to a 100%. We can represent percentage margins of FrieslandCampina and Retailers on the y-axis of a certain graph against time on the x-axis. In particular, the questions of interest are, which one of the two is making more margins and how farther apart are their proportions from the total share?



(a) Lollipop chart to compare percentage FrieslandCampina and Retailer margins.



(b) Smoothed line graph for comparison of percentage FrieslandCampina and Retailer margins.

Figure 6.12: Two approaches for the comparison of percentage margins of FrieslandCampina and Retailers.

In figure 6.12(a), we illustrate how a Lollipop chart can be used. In function, this chart is similar to bar chart but visually, it consists of a line anchored from the x axis and a dot at the end to mark the value. We propose a slight adaptation of a traditional lollipop chart such that the line segment can link two marker values (one each for representing margin value of FrieslandCampina and Retailers) that are suspended in the X-Y space rather than originating from the x-axis. By plotting FrieslandCampina's and Retailers' margins as a lollipop chart, we can analyze how close or far are the two parties with their own share of the margin pool. The length of the line linking the two data points or markers can convey this aspect. The distinct colors of the data markers make it easy to see alternating patterns of who makes larger percentage of the total margin pool. However, the downside of this approach would be apparent in cases where the margin percentages are not very far apart. The center location of the circular data markers are difficult to judge, because closer values would mean that the data points would overlap thus making the comparison imprecise. This would also result in an unnecessary clutter and would make the use of data annotations on the graph compulsive.

Another suitable approach is shown in figure 6.12(b) where a smoothed line graph has been showcased. This type of visualization not only communicates the trends better, but also provides a clear view of instances where the margin shares becomes equal (50%-50%) as the line traces converge at a point. With a continuous line graph, it is easy to follow a chosen trail or path from the starting point to the end and still compare how far apart two points (percentage margin values) are. It is also easy to spot through the distinct colors, where a switch in the trends of the two, occurs. However, as seen in the figure, this visual technique shows values at time points which are not actually available. This is because of the line that connects two adjacent data markers.

After considering several possible visualization methods and discussing their usage in our context, we have a set of candidate design ideas that could be chosen as visual techniques for the visual analysis part of the interface. Let us further continue with the requirements and design principles to be chosen for the remaining set of user tasks.

6.7 View for comparison between different scenarios

Design for user task T7: While outlining design possibilities to address user task **T4**, we saw how comparison of variables *within* the simulated scenario is important. On the other hand, the interface must incorporate a functionality where the state of the current simulated scenario can be saved and compared to another scenario with different price settings. In brief, with user task **T7**, our design should also be flexible enough to allow *between*-scenario comparison. The user should not be forced to remember the previous simulation results for comparison to another set of parameter selections, and compare in his head despite of having a framework that is capable of enabling visual comparison.

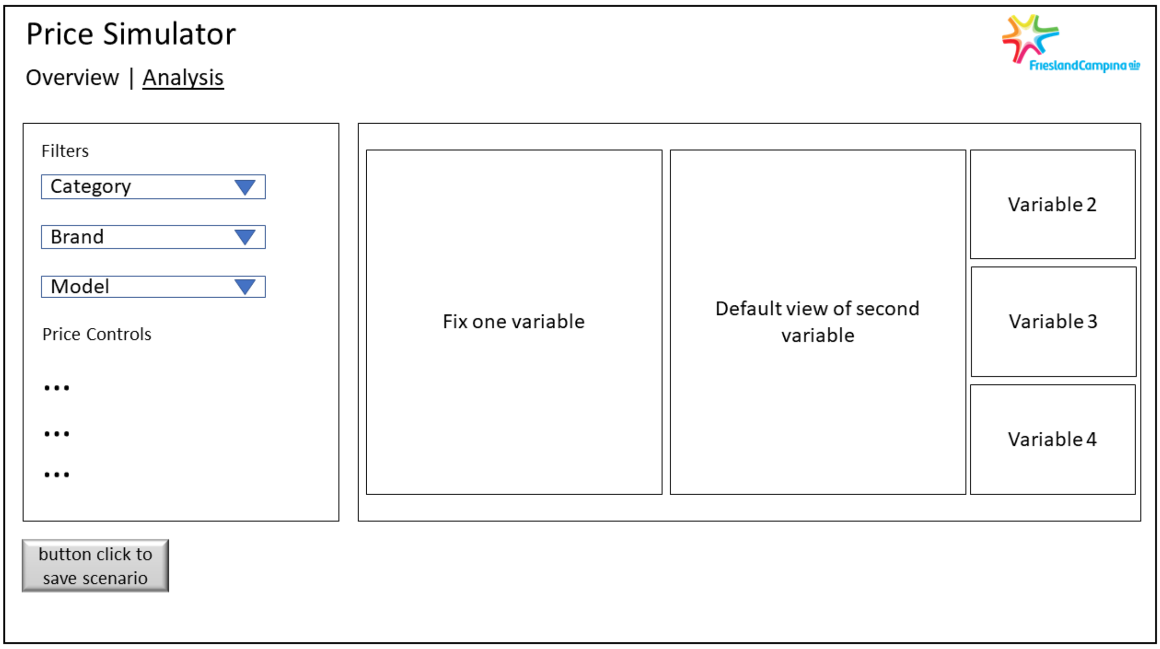


Figure 6.13: Button click functionality to the interface for allowing the users to save their current simulated scenario and continue to simulate other price scenarios depending upon their choice.

In the figure 6.13, we propose providing a button that enables saving a record of the current scenario simulated with certain parameter settings. The next design decision we need to make considers how to display this saved scenario for enabling ease in comparison. The discussion about this aspect of our design decision follows now.

In figure 6.14, we try to explain the proposed idea with the help of a mock-up. When the user initiates the button click to save the current simulated scenario, a duplicated state of the scenario is appended underneath the first horizontal content container, containing both the filter and price panel, and visual contents. The advantage of this method is that the positioning of the two scenario panels, one below the other, can enable the user to compare them. On discussing this functionality with the potential users of the interface, they proposed including a feature where they can give a custom name to the scenario to be saved.

To meet this user request, our design can be made to accommodate a user-input component where the user can input the name they wish to give to their scenario, as seen in figure 6.14. This can be placed at the top of the visual content container with a default instruction message ("give a name to your scenario") as a hint to the user. In this way, when the state of this scenario is to be saved, the custom name given by the user can also be retained for later reference.

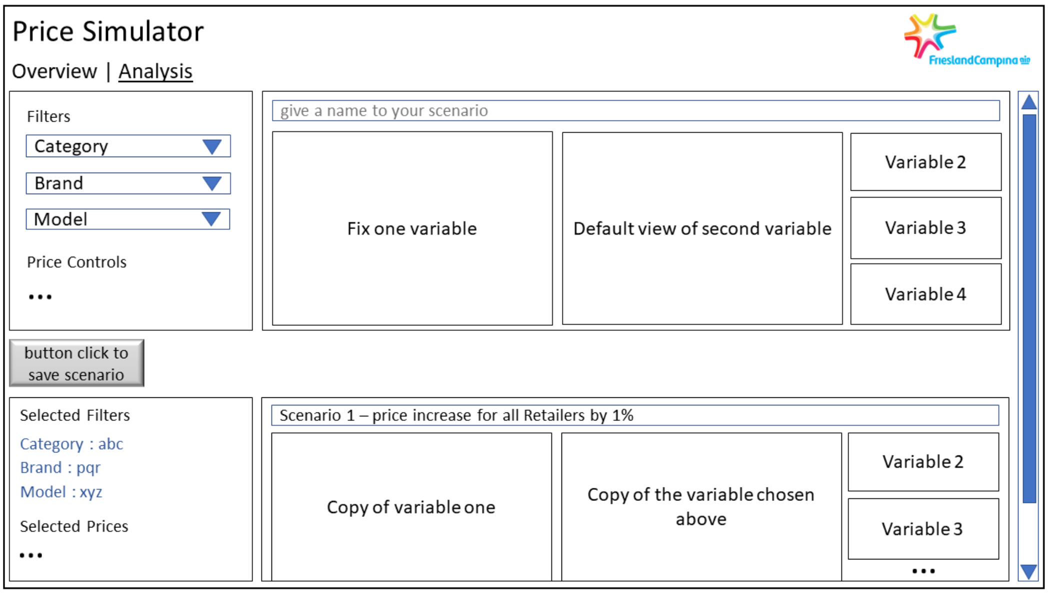


Figure 6.14: *Between*-scenario comparison by appending the saved state of the current scenario exactly underneath the first horizontal content container. The users can also provide a custom name to the scenario before saving it.

6.8 Visualizing per-retailer contributions

Design for user task T8: Let us state this user task more formally with the help of an example for a clearer understanding of its importance. Suppose we increase the price (PTC) for both Retailer 1 and Retailer 2, by 1%. As a result, we see a decline in the total percentage Sales Volume from the baseline, by 4%. Now, the important question that arises is, how much do each of these retailers contribute to this overall decline in Sales Volume? This question emerges from the fact that the price elasticity for each retailer is different. In light of this research question, the user would like to identify each retailer's contribution to both change in Sales Volume as well as Margins. We emphasize and restrict this research question to only these two variables because eventually, the aim of devising pricing tactics should be able to balance the trade-offs between these two variables. Here, we propose a simple visualization method to see the influence of each retailer, when their price is changed.

Figure 6.15: Bar graphs with retailer names on the x-axis and percentage contribution to Sales Volume on y-axis. The color scheme red represents a retailer's contribution to the decline in total percentage Sales Volume while the green color scheme shows a retailer's contribution to the increase in total percentage Sales Volume.

Figure 6.15 shows two bar graphs with Retailer names on the x-axis and percentage contribution on Sales Volume on the y-axis, as a mock-up. In figure 6.15(a), the red colored bars indicate that at the given price setting for the three retailers, each of them contribute negatively to the total percentage change in Sales Volume. Meaning, each of the retailers cause the total percentage Sales Volume to decline. The red color gradient assigned to these bars exemplifies that Retailer 1's contribution to the overall Sales Volume decline is the highest of all. This can be typically observed when retailers increase their price to consumers.

In figure 6.15(b), we see that at a given price setting for the three retailers, while retailers 1 and 2 contribute to an overall decline in the Sales Volume, retailer 3 has an opposite, positive impact. Meaning, the third retailer's price change is causing the total percentage Sales Volume to increase. Such a scenario can be typically observed when retailers 1 and 2 increase their price to consumers, and retailer 3 decreases its price. In this way, we illustrate how this visualization method could support cases where both positive and negative contributions of the retailers in correspondence to their direction of price change are to be presented.

Chapter 7

Implementation and Results

In this chapter of the report, we briefly explain the implementation of our design principles and discuss the results. The elements of the developed visual analysis tool that we call Price Simulator, is demonstrated here.

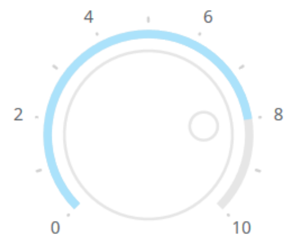
7.1 Implementation

In this section of the report, we aim at discussing in brief, things that could not be implemented due to either the limitation of the development framework or data incompleteness. As stated previously, the framework chosen for the development of the visual simulator is Dash by plotly using Python programming language. As rich as the framework is with its support of several component libraries and in-built functionalities for making design and visualizations highly customized, it lacks the support of some really powerful visualization methods and features. This is mainly because the framework is under a constant development and not all the bugs are resolved by the developers yet.

In line with the design principles laid down for each user task, let us express where our design ideas could not be carried forward to the implementation stage, and the alternatives that were opted for, to overcome these limitations. In our design for user task **T2**, we proposed the use of a three dimensional data table (figure 6.4) in order to structure and organize the range sliders for adjusting the price percentage. Dash's *DataTable* component library is useful in creating interactive table components but creating complex datatables with more than one dimensional attributes is not possible. Our idea of embedding range sliders inside a Dash DataTable thus cannot be implemented.

Next, in order to come up with a robust design for the price sliders to achieve user task **T3**, such that the horizontal space used by the price input components can be minimized as much as possible, many components that are supported by Dash, were considered. Other than using a range slider that shifts to right for increasing the price percentage and left for decreasing the price percentage, the use of a knob (shown in figure 7.1(a)) or a numeric input (shown in figure 7.1(b)) were also considered. Range sliders take up a lot of horizontal space and this problem can be overcome by using components that occupy smaller width. In case of the Knob components provided by one of Dash's component libraries, a negative range of values is not supported. In our context, this would mean that the user can only increase the price percentage from 0 to 5% but cannot lower it. The possibility of using the Knob component for turning the price percentage was thereby discarded.

A better and rather simplistic solution could be to use a numeric input component where the user can simply interact with the upper and lower arrows to higher or lower the price. At the same time, it would also use lesser vertical height in our filter and price panel, thereby reducing unnecessary scrolling. However, the use of this component was not compatible with the way our



(a) A knob component that can be turned to a value between some range, suitable for selecting the input price parameter.

(b) A numeric input component that can be set to a value between a given range, suitable for setting the price parameter values as an input.

Figure 7.1: User input components, apart from the proposed usage of range sliders, supported by Dash framework suitable in our context of inputting percentage price increase or decrease.

Figure 7.2: Dash RangeSlider component that can be user per retailer to support sliding among different price percentages with a defined step-size.

(Python) source code was written. As its implementation could lead to major inconsistencies in the source code, the possibility of its was ruled out. We therefore used Range sliders for price as a user input. Although it uses a lot of horizontal space, the clear annotation of the step-size, i.e. -5% to 5%, makes it easy for the user to realize the possible end-to-end range of the price as input.

Among all the design ideas discussed under **Design for user task T6**, Horizon chart was seen as a promising way of visualizing negative and positive change of a metric from a reference, baseline or horizon. However, Horizon chart is not one of the in-built visualization methods of graphing techniques provided by the Dash framework. Therefore, the design choice of combining bar graphs and a Stream graph to address the user task seems as the most logical solution for implementation.

Further, under the design discussion of this user task, we also discussed possible graphs to visualize Net Sales Value and Margin KPI's. However, one major hurdle in developing a complete visual simulator that is capable of solving our hypothesis 5.3 was experienced with the lack of availability of PTR data. The retail price per liter and total costs per liter data was not available for inclusion in our simulator due to which, the variables dependent on these price values, namely, Net Sales Value, Total Cost Of Goods Sold and Retailer and FrieslandCampina margins could not be modeled or included in the web application.

Despite of this limitation, the design principles and concepts shown and used for implementation are flexible to support the missing data for visualization, as soon as it is made available in the data pipeline API. This also means that our current price simulator takes only two out of the three price parameters as an input, the Price To Consumers and Competitor's Price. Thus we will see how our design was adapted due to these limitations, in chapter 7.

Finally, achieving user task **T8** was critical in terms of its implementation. In order to learn the contribution of each retailer to the total change in Sales Volume when all of their prices is set to a certain value, it was first necessary to compute their own individual contribution to the total change in Sales Volume at each price value. Let us explain this with the help of an example. Suppose that retailer 1 changes its price by x% and retailer 2 changes its price by y%. Due to this, the total percentage change in Sales Volume is z%. Now, in order to compute how much each of these retailers contribute individually to this z% change in Sales Volume, we need to calculate the following:

$$\text{\% contribution of retailer 1 to change in Sales Volume} = \frac{\text{\% change in Sales Volume due to x\% price change, keeping y constant}}{\text{z\%}} \tag{7.1}$$

$$\text{\% contribution of retailer 2 to change in Sales Volume} = \frac{\text{\% change in Sales Volume due to y\% price change, keeping x constant}}{\text{z\%}} \tag{7.2}$$

In other words, we are wanting to compute a part-to-whole relationship, where the % impact caused on Sales Volume due retailer 1 and retailer 2's prices are two *parts* that must sum up to form the *whole*, the total percentage impact on Sales Volume.

However, with the data pipeline available, this was not possible to compute in an automated way because the designed API does not slice the forecasted changes in Sales Volume and Sales Value per retailer slider. Therefore, the individual contribution of a retailer's change in price to the total change in Sales Volume and Sales Value cannot be computed directly. In order to accomplish this task, the contributions of retailers on the Sales Volume was computed manually, for only one model, and is represented in on the user interface to convey its importance and significance.

7.2 Results

In chapter 6, we talked about several design principles in order to achieve the user tasks. By carefully understanding the use of all tasks, we tried to list a number of functionalities and techniques that can be incorporated in the tool for the user to engage in an exploratory analysis, that which is open-ended and not very restrictive. However, not all the design propositions can be fulfilled, as discussed in section 7.1 of this chapter. We are now left with a concise set of design ideas which we implement in this section for developing the visual tool, that we call the Price Simulator. The objectives of the visualization analysis tool include its ability to visually explain our hypothesis with a narrowed scope due to data limitations. Moreover, the user's analysis journey should ensure that the he or she gets a mature understanding of the underlying parameter space. Lastly, it would be a plus if the users can take complete control over the tool and that he or she can do this independently. The results of the implementation of our design principles, as shown and discussed in this chapter, will help us evaluate if these objectives are met. In the following portion of this chapter, we present the usability and results of the implemented *Overview* and *Analysis* pages.

7.3 Overview page - Price Simulator

The overview page is the landing page of Price Simulator, where the user lands when the web application's user interface is first accessed. The implemented version of it is illustrated in figure 7.3, where it is shown how the user tasks **T1**, **T2** and **T3** can be achieved. As per our design concepts, the filtering tasks can be undertaken through the left section of the page, and visual insights can be drawn from the right section of the page. Let us discuss the components on the left of the page.

- **Filters:** This section comprises of single selection drop-down menu selections for Category, Brand and Model. The three drop-down menus are positioned one below the other in the

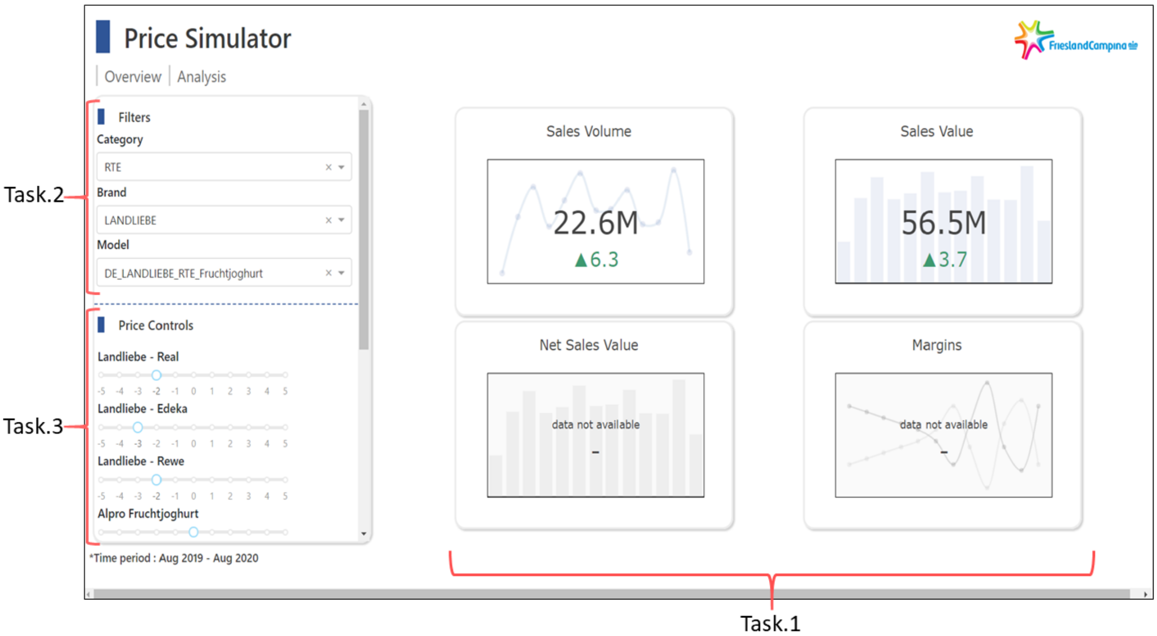


Figure 7.3: Overview page of the Price Simulator. The first three user tasks can be achieved through this view of the interface.

order of a typical product hierarchy. First the Category of products should be selected. In the figure, we see RTE, which is "Ready To Eat" as the selected product category. Based on this selection, the Brand drop-down menu is updated with related options. We see Landliebe as the selected Brand in the figure. Lastly, the Model drop-down selection menu is updated with a list of product-cluster models which are available under the selected Brand. We see the Fruchjoghurt model selected for this attribute in the image. When the Overview page is first loaded, the drop-down selections seen in the image are set at default.

- **Price Controls:** Based on the filter selections, the section under the Price Controls heading of this panel is auto-populated with price sliders corresponding to the retailers which sell Landliebe's Fruchtjoghurts. In the figure, we see three retailers under this filter settings, Real, Edeka and Rewe. Following these, a slider for each of the competitor brands that sell similar product to consumers, is available in our interface. By annotating the slider's range values with numbers (-5% to 5%), the user is kept informed about his selection. While the step size of the slider's range is 0.5, this is not evidently clear in the view. This means that the user would not know immediately that the price could also be tweaked by 0.5%. To overcome this, a visual hint informing the user about the step size of the price slider could have been provided.

The elements on the right side of our design space are in accordance with our design decisions. We see four floating blocks, one for each of the four KPI's. Within these block components, we see a blurred, overview trend graph with the total baseline value superimposed on top of it. These trend graphs are efficient in providing an abstract view of the trend of the variable over time, while highlighting the total baseline value to grab the user's attention. When the price settings are changed, the graphs animate in real-time. This provides the user with a real-time feel of the drifts in scenarios with respect to price change. A functionality where these graphs update as the slider values are dragged, and not after the slider's hold is released, has been provided on the interface for enabling quick and immediate interactions. The delta indicators help the users identify if the price change caused a total uplift or downfall from the baseline value. The green or red color of the indicator, and the direction to which it points, helps in drawing such insights.

When the web application is first loaded, the price sliders are set to a 0% price change by default. This is to give the user a basic view of the current scenario. We show the trend in Sales Volume over the year with the help of a Line graph so that the peaks and troughs over time become prominent. When the user observes some unusual trends, the idea is to switch to the Analysis page to unlock the hidden details. Similarly, for the Sales Value, we chose a bar chart representation in this view to associate the height component with the revenue through product's consumer sales. We do not discuss the remaining two charts in detail as they have been displayed as dummy.

The design is capable of summarizing all variables and providing a clean presentation through their discrete positioning. The representation of the overview in this way makes our visual analysis tool appear like a web reporting tool. This means that the Overview page as a whole, can be used in an early part of the data analysis process and serves to present data in a visual and slightly interactive way for further data discovery or data analysis.

Although the filter panel can be smoothly accessed through scroll-up and scroll-down actions, it might be difficult for the user to transit between the first and last slider component in the list. This part about the design could be reconsidered. With the framework's limitation, this was the best possible solution that could be achieved in the limited time frame of the project's development. A note added to the footer of the Overview page's view informs the user about the time frame upon which the analysis is based. After a higher-level abstract overview, the user can navigate to the Analysis page through the selection of *Analysis* page-link provided at the top-left of the web page.

7.4 Analysis page - Price Simulator

In figure 7.4, we see the layout and view of the implemented Analysis page. All the components within this illustration are explained in the following sections of this chapter.

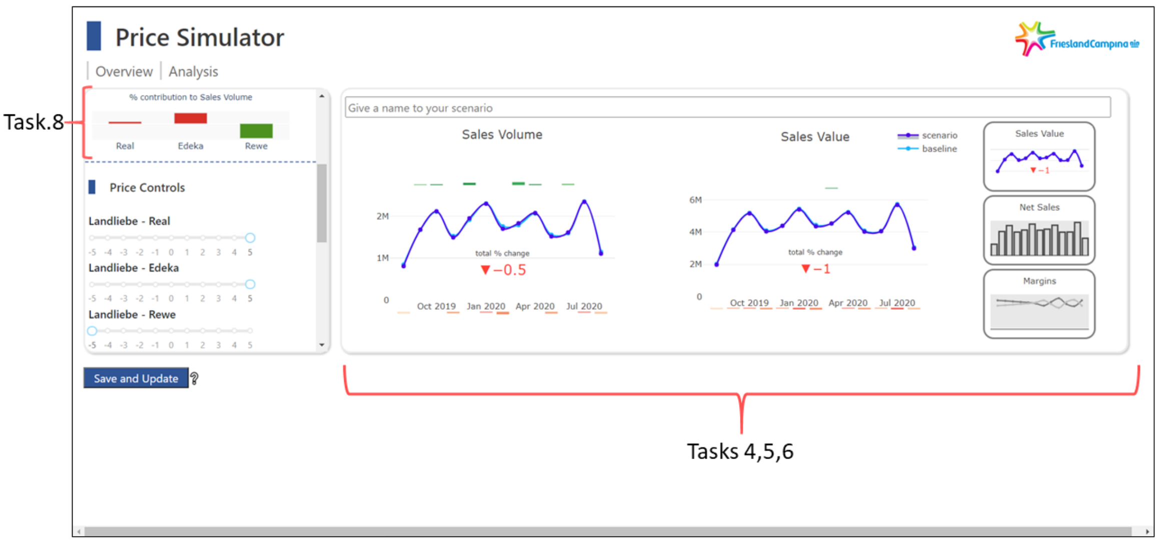


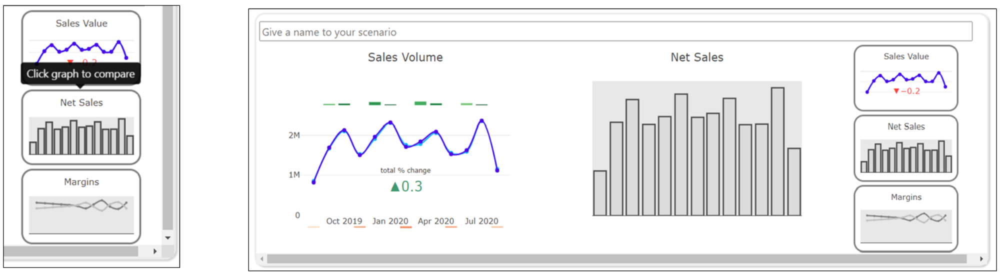
Figure 7.4: The Analysis page of the Price Simulator when the user interacts with the price settings. The visualization tasks that can be achieved have been annotated.

7.4.1 Selection of variables for comparison

On the right section of the Analysis page seen in figure 7.4, we cage all the visual components inside the body of a parent container to achieve our proposed design. As per the design choice we made for user task **T4**, we fix the visualization graph for Sales Volume at the first position within the parent container and place the Sales Value graph right next to it. This means that when the user lands to the Analysis page, the Sales Volume and Sales Value are the two default juxtaposed graphs that can be used for direct comparison by the user. Had the retail data been available, the default choice of the first, fixed graph and the second default displayed graph, would have been Sales Volume and Margins, respectively. This is because the primary aim of the visual analysis tool is to compare these two variables against each other in order to balance their trade-offs when the price is changed.

The small multiples to visualize Sales Value, Net Sales Value and Margins are also a part of the view, as per our design principles. The first two enlarged graphs of fairly large resolutions provide the ability to observe and gain insights into the details of the data, for a deep-dive analysis. At the same time, the small multiples provide a minimalist yet clear view of the variables' values and trends.

In figure 7.5, we show the "Click to compare" feature of the price simulator. When the user hovers over the small multiples, a visual hint informing them about the ability to click the graph and enlarge it for comparison, appears. In the figure, we see that on clicking the Net Sales Value graph among the small multiple views, it is expanded and positioned next to the Sales Volume graph for direct comparison. In this way, we provide more flexibility to the tool in terms of choosing variables for comparison.



(a) Visual hint for "Click to compare" functionality on hover.

(b) Juxtaposition of the Net Sales Value graph against the Sales Volume graph to enable direct visual comparison of the two KPI's.

Figure 7.5: Juxtaposition of the Net Sales Value graph against the Sales Volume graph when the user clicks on the former's small multiple view to enable direct comparison.

Let us now move forward with presenting how this user interface implements the design principles of visualization techniques for a deep-dive analysis to accomplish the user tasks **T5** and **T6**.

7.4.2 Visualization techniques

With the use of Stream graphs seen in figure 7.4, the trends of baseline value and the forecasted scenario can be visualized. We see seasonal peaks and troughs in the sales volume and value, every third month. As seen in the figure, the scenario line trace does not deviate much from the baseline trend as a result of which, the two line traces remain superimposed throughout almost all the time points. This is because, for smaller changes from the baseline values, the deviations would not be noticeable. In this case, the grey color used for filling the area between the two line traces as an indication of deviation, is also not visible. While the distinct colors of the data markers and lines help in spotting out deviations, the negative and positive bars on the bottom and top axis of the stream graphs, also help in grasping the deviations visually. With such a visual representation, the user would be able to associate the height and color intensity of these bars to the amount of difference between scenario and baseline values, either positive or negative. In the figure, we also see a situation where the change in price settings, causes the scenario value to increase from the baseline over some months and decrease over the other. This aspect makes our visualization scalable to support unusual and interesting behaviours. The Δ indicator positioned at the center of each visual graphs, summarizes the overall percentage change of the variable due to price change. Its positioning under the stream graph curve is suited to avoid interruptions with the bar graphs at the bottom or top. Our design decision of combining three graphs under single representation by using time series visualization techniques is able to capture the multiples user tasks efficiently at one given point of the whole design space. The resulting implementation is also aesthetically appealing.

The challenging part about implementation of such a visualization was creating graphs over a shared x-axis. To achieve this, critical decisions had to be taken in order to divide the vertical height and scale of the entire graph among three component y-axis values.

Since the chart type for both Sales Volume and Sales Value is kept same, a common legend for the two has been positioned at the top-right corner of the view. A legend for the bar graphs representing absolute differences have not been added to the view because of the framework's limitation of showing legend markers for graphs with colorscale gradients. For the visual illustration of the Sales Value plot among the three small multiples, we chose to provide only the Stream graph with Δ indicator representation of data in order to avoid complexity and clutter within the limited space. The Net Sales Value and Margins plots have been intentionally made grey as these do not represent the actual data but only imitate how the visualizations could look like vaguely.

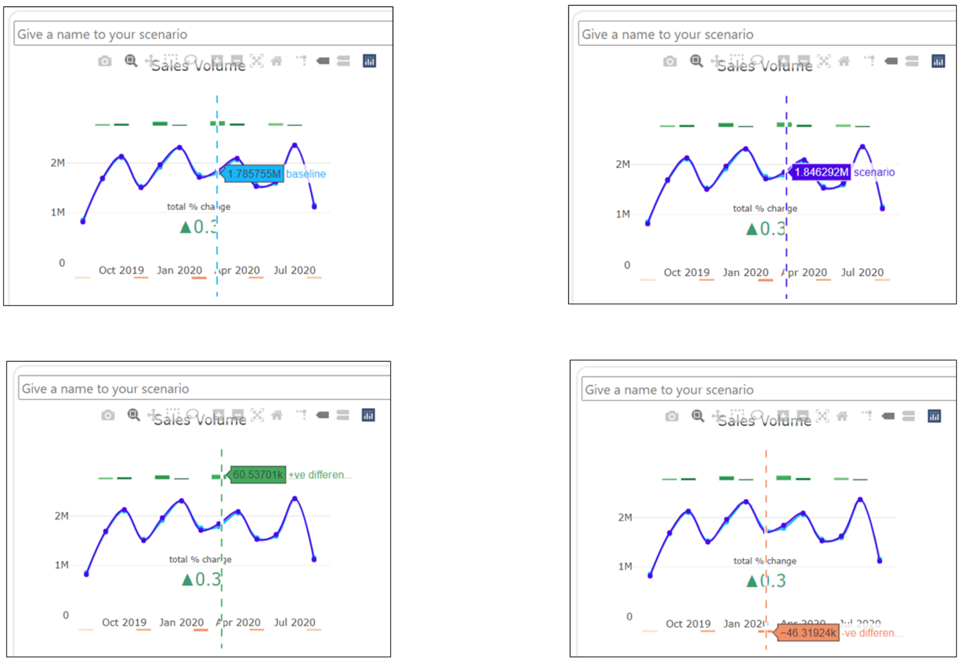


Figure 7.6: Hover-based tooltip display of data values along the line traces and bars.

In figure 7.6, the hover-based tooltip functionality has been illustrated. When the user hovers the cursor along the data points or the graph traces, the numeric value at each point appears for details. A special feature of displaying a dashed line that spans the entire vertical height of the graph, stream graph and bar graph combined, has been provided with the tooltip functionality. The advantage of such a vertical dashed line is that at the point of hover, the values corresponding to the same time point can be located on the other visual components of the plot (bars, lines as well as the month on the x-axis), at the same time.

One major drawback due to the framework's limitations was that it was not possible to introduce linked hover functionality that would allow the user to hover over all the graphs at the same time. This was because, in the context of our implementation, the framework could not support this functionality.

Let us now explain the implementation of user task **T8**, as annotated in figure 7.4.

7.4.3 Visualizing per-retailer contributions

In our context, the user task **T8**'s design and implementation falls completely under the choice of the designer. In order to understand the percentage contribution of each retailer on the total percentage change of Sales Volume as the price per retailer is changed, one of the simplest approach was to create a bar graph that visualizes per-retailer contributions. We position this visual component inside the filter and price settings panel, and exactly on top of the per-retailer sliders. This is because, while the user tweaks price settings for the retailers, the retailer's individual contribution to the consequent change in Sales Volume can be immediately prompted to the user to enhance their analysis.

As seen in figure 7.4, retailers Real and Edeka contribute negatively to the total -0.5% decline in Sales Volume. This is because, the prices for these two retailers have been increased to 5%, and as per our hypothesis, this would mean a decline in the Sales Volume. However, it is interesting to notice that at a 5% increase in price for both Real and Edeka, the latter's contribution to the overall decline in Sales Volume is much higher than the former's. The height of their corresponding bars help us deduce this information. On the other hand, if Rewe decreases its price by 5% in this scenario, it actually contributes positively to the total change in Sales Volume, i.e. it contributes to an increase in the total percentage change in Sales Volume. Again, the hypothesis holds true here. In other words, we could also say that the price decrease of Rewe, makes the overall 0.5% decline in Sales Volume, less negative. This positive behavior is seen in the bar graph, where the green bar represents an opposite, positive contribution to the overall decline in Sales Volume. It can also be deduced that Rewe has the highest contribution among all the three retailers to the way the Sales Volume changes from the baseline value.

The colorscale for this bar graph visualization could also be set to a diverging range of color intensities to make the relative contributions explicit. A possible alternative to the bar graphs could be Heat maps, to depict retailer contributions through colors and color intensities. The heat map corresponding to each retailer could also be placed right next to their price sliders.

Moving on with our results, we now discuss the results of implementing the design choices for accomplishing the user task **T7**.

7.4.4 Comparison between different scenarios

As per our design choices, the *between*-scenario comparison has been implemented in the Price Simulator and the results can be seen in figures 7.7 and 7.8. In order to save or capture the visualizations produced by the user in the first scenario, we have proposed a method that saves the current state of the simulations in a duplicated form.

When the user clicks the button "Save and Update", the exact copy of the filter and price settings panel, and the visual output container, is appended below the upper set of containers. Before saving the simulated scenario, the user can provide a custom name to it for later reference or use. This provision of the interface is also inline with our proposed design idea. In figures 7.7 and 7.8, we see that the duplicated filter and price settings panel is static and non-interactive. It records and displays the settings at which the filters and prices were set to, in a list view. On the other hand, the right visual section is rather interactive in a way that the user can still use the "Click to expand" functionality to toggle between the three small multiples and enable *within* as well as *between*-scenario comparison among variables.

Figure 7.7: The save functionality of the Price Simulator for recording a simulated price scenario. The saved record is appended underneath the previous set of scenario containers. The left section of the saved record lists the selected filter and price settings. The right, visual section of the saved record, is saved in the same state and is interactive.

Figure 7.8: Visual comparison of Sales Volume graph against different KPI variables through the click to expand feature. After saving the scenario, it is still possible to toggle between the small multiples for *between*-scenario comparison.

By positioning the duplicated state of the scenario below the former scenario, the user can update the former scenario with different settings to compare it with the saved scenario. The two different scenarios can easily be compared one below the other, as they fit exactly into the total vertical height of the web page. This comparison technique makes it easier to capture similarities and differences between different price scenarios and analyze them.

The method implemented for visual comparison by this technique is however, not the best way to compare for instance, more than two scenarios. A possible way of making up for this limitation could be creation of a third view in the interface, apart from Overview and Analysis pages, where the graphs produced for each of the simulated scenarios, could be organized in different layouts specified by the user, such that multiple scenarios could be compared at a time. Moreover, the interface could have been enhanced if the duplicated set of containers (both filter, price settings panel and visual container) could be linked for an independent scenario simulation against the former scenario. The complexity of this feature's implementation was quite high and could not be explored in-depth.

In general, we have been able to implement all the user tasks by exploring and using several visual techniques. All the components of our design have been included in the interface with some adaptations. As a proof-of-concept, our visual analysis tool incorporates the design principles that help in achieving user tasks in a simplistic yet interactive way.

Chapter 8

Evaluation

In this chapter, we try to answer our research questions after the implementation of our Price Simulator. A short user evaluation has also been provided to evaluate our results from the perspective of the potential end-users.

8.1 Research questions revisited

At this point in the report, we have seen how the implemented user interface looks like, and the aspects in which it supports user tasks and questions through visualization and design concepts. Let us now evaluate how the problems we were aiming to solve initially are being solved by our product, and if we are able to solve them at all.

1. Which are the variables that contribute to a consequent change in Revenue KPI's when the price of a product or product-cluster is changed, as a function of price elasticity?

   Answer: By creating the margin computational model explained in chapter 4, we were able to map the dependencies among variables. It is clear how to compute the change in Revenue KPI's when the price inputs change. This created a foundation of our price simulator where these calculations need to be modeled.

2. How does price change impact the volume sold to retailers as a function of their respective price elasticities?

   Answer: As per the hypothesis, we are able to show that an increase in the Price to Consumer, the Sales Volume declines. In contrast, a decrease in price increases the Sales Volume. While this is modeled in the black-box price elasticity model, our aim of leveraging visualization methods and techniques to communicate this to the users who do not understand the nuances behind the price elasticity model, stands fulfilled. In the beginning, we also claimed that each retailer reacts differently to a change in price. This aspect, termed as the contribution of retailers to the change in total Sales Volume, has also been accounted for by computing and conveying it to the users through visualization.

3. What happens with the (total) FrieslandCampina margins and profitability when the price of a product-cluster is changed?

   Answer: Due to the lack of retail sales data, we were not able to explore and implement this, highly important detail in our visual simulator. In order to learn the sales and profitability trade-offs better, it is essential to incorporate Margins into our framework and tool for a complete and unbiased understanding of the concept. In principle, pricing tactics can never be pursued without analyzing impact of price change on both sales volume and margins.

4. What is FrieslandCampin's margin compared to Retailers' margins for a given product-cluster?

   Answer: Due to the reason stated above, this question could not be answered through our implementation as well.

5. What would the impact on FrieslandCampina's sales volume and margins be when their competitors in the same product segment change their price to consumers?

   Answer: We are able to observe the impact on Sales Volume when the competitors selling a comparable product, change their price to consumers. Typically, with the help of this question, it is possible to predict how external factors impact the sales and profitability for the organization. By providing sliders to tweak competitor prices, the consequent impact on Sales Volume and Margins could be observed. Since the project's scope did not involve explicit competitor analysis, special attention was not given to this aspect.

6. Which kind of techniques would allow users to undertake a comparative analysis of different price scenarios?

   Answer: In order to chose efficient visualization techniques, factors like the type of data available, the data relationships to be unfolded, and the user tasks to be achieved were investigated. In our project, instead of using complex visualization and comparison techniques, the simpler ones were chosen and integrated to meet the purpose. Superposition of baseline and forecast values was effective in visualizing the deviations. Using small multiples and large singles for our design space facilitated comparison and presentation. Other than this, the juxtaposition technique for visual comparison was also found to be useful.

7. Finally, how can such analyses be made more intuitive and insightful than the traditional means of using Microsoft Excel or business intelligence tools, for instance, to carry out data-driven investigations?

   Answer: The answer to this question is rather implicit. Creating visualizations by first recognizing what the user wishes to analyze from the data, adds value and meaning to the end result. In attempts to develop a visual analysis tool that is capable of meeting all user requirements and incorporating all user tasks, the designer of the interface resorts to the development of a new user interface from scratch. In this way, a customized, personalized and fully equipped tool, in terms of its support of all the desired features, can be provided to the end-user. The traditional platforms lack the ability to support complex analyses that require intuition more than computational granularity. Existing Business Intelligence Tools provide only a restricted choice of visual graphs and methods for representing data due to which, the developers may either end up using more complicated graphs and charts than what is actually necessary, or too simple ones that do not bring out the intrinsic features in data. In our context, due to the requirement of dealing with numerous parameter interactions at once, a platform which can handle complexity and provide more sophisticated functionalities was essential and therefore helpful.

8.2 User evaluation

With seven of our potential users within the Digital Media and Insights Team of FrieslandCampina, a demonstration and feedback session after the development of the Price Simulator was conducted. After demonstrating the usability of the tool in general, the participants of the user evaluation test recorded their feedback in the form of answers to a given set of questions. The questions that were asked, are listed in table 8.1. While the first six questions were a "Yes/Somewhat/No" type of questions, the remaining four were of subjective type.

Question 1	Is the visual simulator capable of answering business questions, such as, how does price change impact Sales Volume and Sales Value?
Question 2	Do the visualization methods used support your analysis?
Question 3	Is the tool user-friendly? (Can you use the tool independently without explicit guidance of how to take the necessary steps?)
Question 4	Does the tool provide sufficient support for comparing different variables at the same time?
Question 5	Does the tool provide sufficient support for comparing different price scenarios?
Question 6	Are you satisfied with the choice of colors, legends and tooltips?
Question 7	Do you find this visual analysis tool useful for simulating price scenarios or would you rather use, for instance, Microsoft Excel for computations and Power BI for producing visualizations? Why?
Question 8	What could be improved in the tool from data visualization perspective?
Question 9	What kind of functionalities does the tool miss, if any?
Question 10	What did you like the most about the tool?

Table 8.1: Listed are the ten questions included in the user evaluation test for seven of the potential users of the Price Simulator developed.

For *Question 1*, seven out of seven recorded *Yes* as their response as they thought that the visual simulator is capable of answering business questions with regards to Sales Volume and Sales Value as the two KPI variables. Six out of seven users thought that the visualization methods implemented did support their analyses, while one of them opined that it somewhat supported them (refer *Question 2*). One of the most important aspects kept in mind while designing the tool was put forth as a question to the users, whether they think they can use the tool independently without explicit guidance. Seven out of seven thought that they could (refer *Question 3*). For *Question 4*, six out of seven recorded answers were seen to be in favour of the tool being able to compare different variables at the same time. One out seven expressed the lack of presence of Net Sales Value and Margins variables for comparison, which at the moment are not available for comparison in the tool due to data unavailability. All the seven respondents agreed that the tool is capable of providing a visual comparison method of price scenarios (refer *Question 5*). While most of them were satisfied with the use of colors, legends and tool-tips in the interface, one of the responses proposed that a color theme that could incorporate FrieslandCampina's identity could also have been considered (refer *Question 6*).

For the open-ended *Question 7*, all the participants opined that developing tools like the Price Simulator, provides a user-friendly framework to users that can also handle complexity. At the same time, it is less error-prone as compared to commercial spreadsheets like in Microsoft Excel. However, some argued that developing dashboards and visual analysis tools from scratch also require heavy time investment which would be easier with existing Business Intelligence tools, which is why the answer to this question is rather situation-dependent. When asked about what could be improved in the tool as per *Question 8*, some of the users put forth the idea of introducing visual methods to see the impact every variable has in itself on the model, keeping other variables constant. This points to the use of influence or contribution measure of every variable instead of few, on the four KPI's. What the users missed seeing in the tool (referring to *Question 9*) was, the comparison of Sales Volume with Margins. Apart from this, the users thought that adding a module for recommending optimal price point based on the margin and price elasticity, would be interesting to see. Some also pointed out that the simulation environment could also support price change of two or more product-clusters (not just retailers) at the same time. For *Question 10*, the users were happy about the level of interactions supported in the tool, specially the flexibility of clicking on variables to expand their view for direct comparison and analysis. Some also thought that the tool will help them make computations easily, through the simple and clean visuals provided.

In general, the evaluation helps us conclude that the data limitation inhibits the inclusion of important variables such as Margins and Net Sales Value. Once these are available, the users' recommendations of determining optimal pricing by balancing volume and margins could be satisfied. Other than that, the users would like to see impact of price change on not just retailer level, but also at the product-cluster level. The aesthetics and usability of the tool seem to satisfy all the users.

Chapter 9

Conclusions and Future Work

At the beginning of this project, we began with a concept and hypothesis in mind. We studied the concept of price elasticity of demand and learned that the demand of a given product could be price elastic or inelastic. If it is elastic, a small percentage increase in its price causes a significant decline in its demand, and if inelastic, a small percentage increase in its price causes a rather smaller decline in its demand. However, while making decisions about suitable or correct pricing tactics, it is not enough to know the impact of price change on demand. It is equally important to consider the impact on profit margins as well since a price increase, most likely, causes a reduction in the profit margins. This is exactly what our hypothesis was. The objective of this project was to use visualization techniques to pursue goal-oriented pricing tactics where the goal could either be increasing the Sales Volume (demand) or boosting margins. Clearly, there needs to be a trade-off or a balance between the two. However, the computation of margins depends on several other variables such as Net Sales Value and Total Costs to name a few. Therefore, it is important to study the impact of price change on all these interrelated variables for a deeper understanding of the parameter space. Further, there are three price parameters that influence our analysis, the Price to Retailer, the Price to Consumer and the Competitor's price. Each of these, along with the price elasticity model act as inputs to the margin computational model. In order to handle the complexity of the model, given the large number of variables involved, we chose to develop a visual simulator that we call as the Price Simulator. The objective of the visual simulator is to aid users engage in a deep-dive analysis of impact of price change on the Key Performance Indicators Sales Volume, Sales Value, Net Sales Value and Margins of both FrieslandCampina and its Retailers.

The choice of developing a visual analysis tool as a simulator is valid in order to mimic how price could be changed in several ways to meet a strategy or goal. With the data about price PTC of product-clusters per retailer and per competitor and a black-box price elasticity model, we were able to define user tasks or actions which he or she would like to undertake with the help of the developed tool. Once these were defined, the visualization techniques that would help achieve each of the user tasks in a proficient way, were determined. This process was iterative, keeping the users in the loop for their feedback. Time series analysis techniques were majorly explored and used to support the user requirements at best. Apart from providing the visual interface with effective visualizations, ways of providing the interface with interactive capabilities and functionalities have also been explored. Features like saving or recording the visualizations produced for a particular parameter setting (price values for each retailer and competitor), selecting to juxtapose variables for visual comparison and providing users with visual hints to guide their own analysis were some of the most powerful ways of making the web application look like a software-like tool where the user can take control for meeting their desired objectives of the analysis. Our implemented web-based Price Simulator incorporates all the eight user tasks that were determined. As per the user evaluation conducted with the potential users of the Price Simulator, our visual analysis tool is user-friendly and flexible in providing comparison of price scenarios.

As of the current state, the developed Price Simulator helps users validate one part of the hypothesis through visualizations; an increase in the price of a product causes a decline in the Sales

Volume. Due to the data limitations, the tool's support for visual communication of the impact of the price change on Margins remains incomplete. This implies that vital business decisions such as pricing and pricing tactics cannot be fully realized without the missing information about revenue and margins. As a result of this, our research questions about the impact of price change on margins could not be answered. However, our implementation is flexible enough to be scaled up to visualize Net Sales Value and Margins as soon as the data is made available. With some limitations of the framework chosen for development, the overall user interface seems to be quite sophisticated and useful for both higher-level strategic users as well as business analysts or strategists to engage in analysis.

One major recommendation to enhance the visualization output of the project would be modification in the format of the data pipeline provided as an input to the user interface. At the moment, the API and backend black box price elasticity model fetch the forecasted values of Sales Volume and Sales Value when a percentage change on the price sliders is initiated on the front-end. In order to model computations and calculations on the data attributes easily, it would more efficient to fetch and store the forecast values before these are rendered onto the interface. Further, in order to support complex computations such as the margins of the retailers (per retailer margins), the values of all the data attributes in the data pipeline must be sliced per retailer to be retrieved by the interface.

9.1 Future scope

One of the most important future improvements of the existing Price Simulator is to obtain and integrate the retail sales data with the consumer sales data so as to compute Revenue and Margins. Once this data would be available, the price simulator would be able to make our hypothesis evident. As of now, it is quite trivial to observe that an increase in price will cause a reduction in the sales volume. Based on this intuition, the user concludes that reducing the price would boost the sales without knowing the other half of the story, what would then happen to the margins. However, data and visual analytics are capable of enriching such a problem description with more insights by providing means of computing optimal price for a product-cluster. By optimal price, we mean that value of the price, at which both sales volume and profitability are at a peak and should no longer be optimized. Since price change impacts both these variables in an opposite way, the user interface could be provided with a functionality where the users can choose which of the two they wish to optimize, if the objective is goal-oriented. If the user is rather looking for creating a balance between the best possible price at which neither the sales volume is too low, nor the profits made are too small, it could be possible to provide on the interface, an input where the user assigns a weight to both Sales Volume and Margins (for example on a scale of 1 to 10, the user could assign Sales Volume a weight 2 and Margins a weight 8), in order to gain the optimized pricing for the chosen product-cluster as per these weight settings. This could be implemented by methods such as Gradient Descent Optimization which can optimize a value, in our context price, for an output function, in our case Sales Volume and Margins. By computing gradients for each of the price parameter (PTC and PTR per liter), the output function could be optimized for a local optimum. With the help of visuals, the users could be informed about the trade-offs and balances. Moreover, visual hints such as "try changing the price to ...", could be made to appear on the interface when the user interacts with the price sliders to change the values.

Apart from this, on adding promotion data to the data pipeline and inputting it onto the interface, gaining insights about the performance of Sales Volume and Margins in weeks of product promotions would be interesting to see. In this case, year over year comparison of the variables with the help of data visualization techniques, would provide knowledge about (a) the impact of different pricing strategies or tactics (if they were different for the years being compared) and (b) the influence of certain promotion periods during particular weeks for gaining deeper insights into the strengths and weaknesses of business strategies.

List of Acronyms

API Application Programming Interface. 17, 19, 20, 51

COGS Cost Of Goods Sold. 8, 15, 31, 51, 56

KPI Key Performance Indicators. 1, 2, 7, 8, 10, 12, 13, 19, 21–28, 32, 37, 39, 41, 44, 46, 48, 51, 55–57, 62, 63

NSV Net Sales Value. 10, 14, 15, 51, 55–57

PED Price Elasticity of Demand. 9, 10, 51, 55

PTC Price To Consumers. 10, 11, 14, 15, 18, 25, 26, 34, 50, 51, 55

PTR Price To Retailer. 10, 11, 14, 15, 18, 25, 26, 37, 51, 55

RRP Recommended Retailer Price. 6, 51

Bibliography

[1] T. Bodea and M. Ferguson. Pricing segmentation and analytics. 2011. 7

[2] Delloite. Getting Pricing Right - The value of a multifaceted approach. https://www2.deloitte.com/uk/en/insights/deloitte-review/issue-3/getting-pricing-right.html, 2008. [Online; accessed July, 2020]. 4, 7

[3] Delloite. Pricing Analytics - the three minute guide. https://www2.deloitte.com/content/dam/Deloitte/global/Documents/Deloitte-Analytics/dttl-analytics-us-da-pricinganalytics3minguide.pdf, 2012. [Online; accessed July, 2020]. 7

[4] S. V. D. Elzen and J. V. Wijk. Small multiples, large singles: A new approach for visual data exploration. *Comput. Graph. Forum*, 32:191–200, 2013. 62

[5] FrieslandCampina. About FrieslandCampina. https://www.frieslandcampina.com/about-frieslandcampina/. [Online; accessed February, 2020]. 1

[6] M. Gleicher, Danielle Albers, Rick Walker, I. Jusufi, C. Hansen, and J. Roberts. Visual comparison for information visualization. *Information Visualization*, 10:289 – 309, 2011. 5, 27, 62

[7] J. Heer, M. Bostock, and Vadim Ogievetsky. A tour through the visualization zoo. *Queue*, 8:40 – 41, 2010. 4

[8] Economics Help. Pricing Elasticity of Demand (PED). https://www.economicshelp.org/microessays/equilibrium/price-elasticity-demand/, 2019. [Online; accessed April, 2020]. 9

[9] Enamul Hoque, V. Setlur, Melanie Tory, and Isaac Dykeman. Applying pragmatics principles for interaction with visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, 24:309–318, 2018. 4

[10] Daniel Keim, Gennady Andrienko, Jean-Daniel Fekete, Carsten Görg, Jörn Kohlhammer, and Guy Melançon. Visual analytics: Definition, process, and challenges. In *Information visualization*, pages 154–175. Springer, 2008. 17

[11] The Financial Times Limited. Visual Vocabulary - Designing with data. https://github.com/ft-interactive/chart-doctor/tree/master/visual-vocabulary, 2019. [Online; accessed June, 2020]. 28

[12] A. Lunzer, R. Belleman, P. Melis, and G. Stamatakos. Preparing, exploring and comparing cancer simulation results within a large parameter space. *2010 14th International Conference Information Visualisation*, pages 258–264, 2010. 5

[13] T. Munzner. Visualization analysis and design. In *A.K. Peters visualization series*, 2014. 5

[14] Nielsen. Nielsen Analytics Solutions. https://www.nielsen.com/us/en/solutions/. [Online; accessed February, 2020]. 14

[15] U.S. Bureau of Labor Statistics. Horizon Graphs of U.S. Unemployment Rate, 2000-2010. https://homes.cs.washington.edu/~jheer//files/zoo/ex/time/horizon.html. [Online; accessed July, 2020]. 30, 31

[16] Massachusetts Institute of Technology Sloan Management Review. Financial Analysis for Profit-Driven Pricing. https://sloanreview.mit.edu/article/financial-analysis-for-profitdriven-pricing/, 1994. [Online; accessed June, 2020]. 7, 8

[17] plotly|Dash Enterprise. Dash App Gallery. https://dash-gallery.plotly.host/dash-financial-report/. [Online; accessed May, 2020]. 5

[18] Hans-Jörg Schulz, T. Nocke, Magnus Heitzler, and H. Schumann. A design space of visualization tasks. *IEEE Transactions on Visualization and Computer Graphics*, 19:2366–2375, 2013. 4, 21, 60

[19] B. Shneiderman and A. Aris. Network visualization by semantic substrates. *IEEE Transactions on Visualization and Computer Graphics*, 12:733–740, 2006. 5

Appendix A

Impact of Price Elasticity on Revenue KPI's

KPI's	current	new	impact %
Volume	1,000	880	-12%
Net Sales Value	2,000	1,936	-3%
Sales Value (excl. VAT)	2,830	2,740	-3%
Margin FrieslandCampina	0.50	0.70	22%
Margin Retailer	0.83	0.91	10%

Table A.1: Example Calculations on KPI's with dummy data when PTR and PTC are assumed to be increased by 10% each, given the PED is -1.2 for a particular retailer.

Prices per Liter	current	new	impact % (assumed)
NSV/L	€2	€2.20	10%
Base PTC/L	€3	€3.30	10%

Table A.2: Assuming the base prices and calculating the respective increments on 10% increase in both PTR (which is same as Net Sales Value per liter (NSV/L)) and PTC, per liter, as for this example.

As shown in the table A.1, let us consider that the current Volume demanded or sold is 1,000 liters. Given that the PED is -1.2 and the PTR increase is 10%, using the formula 3.1:

$$-1.2 = \frac{\text{change in quantity demanded}}{10\%}$$

$$\text{change in quantity demanded} = -1.2 * 10\% = -12\%$$

Hence, the impact on the current Volume is -12%. Thus, the new volume demanded would be 12% less than the current quantity, which results in 880 liters.

Further, FrieslandCampina's Revenue, i.e. the NSV will be given by:

$$\begin{aligned} \text{NSV} &= \text{Current Volume} * \text{Current NSV/L} \\ &= (1,000) * €2 \\ &= €2,000 \end{aligned}$$

where the value of current NSV/L is taken from table A.2.
With respect to the decline in the volume by 12%, the change in the NSV is:

$$\begin{aligned}\text{New NSV} &= \text{New Volume} * \text{New NSV/L}\\ &= (880) * (2.20)\\ &= €1,936\end{aligned}$$

As per table A.2, if the base price to consumer is €3, then this multiplied by the volume sold (1,000 liters) makes a sales value of €3,000. We now need to calculate the new Sales Value:

$$\begin{aligned}\text{New Sales Value} &= \text{New Volume} * \text{New Base Price}\\ &= 880 * €3.30\\ &= €2,904\end{aligned}$$

This is the Sales Value including Value Added Tax (VAT). The tax on products is a value that is country-specific. For instance, in Netherlands the VAT rate is 6%. In order to compute Sales Value excluding VAT, we divide €2,904 by (1+6%), which is 1.06.
Therefore, the New Sales Value excluding VAT is €2,740. This is important to calculate because this is the Retailer's revenue by deducting all costs and charges they pay in order to calculate their absolute margin.

Now, we can compute the margin earned by both FrieslandCampina and the Retailer. For the former party, this margin is the difference between the Revenue (NSV) and the Cost Of Goods Sold (COGS). Let's assume that FrieslandCampina bears a cost of €1.50 per Liter of production/manufacturing the product. So, for a current volume of 1,000 liters, the Total COGS would be $1,000 * €1.50$ which is €1,500.
Thus we have,

$$\begin{aligned}\text{FrieslandCampina Margin} &= \text{Current NSV - Current COGS}\\ &= €2,000 - €1,500\\ &= €500\end{aligned}$$

The per liter FrieslandCampina margin would be $€\frac{500}{1000}$, which is €0.50.

But now, with the decline in volume, the new Total COGS would be $880 * €1.50$ which makes €1,320 the new Total COGS. This is why the new FrieslandCampina Margin now becomes:

$$\begin{aligned}\text{New FrieslandCampina Margin} &= \text{New NSV - New COGS}\\ &= €1,936 - €1,320\\ &= €616\end{aligned}$$

This means that FrieslandCampina's margin for the sales of this product per liter, increases to €0.70.

In a similar way, we can compute the Retailer's margins. This is the difference of the sales to consumers and the price at which the retailer purchased the product from FrieslandCampina (i.e. same as the NSV).

$$\begin{aligned}\text{Retailer's Margin} &= \text{Current Sales Value(excl. VAT) - Current NSV}\\ &= €2,830 - €2,000\\ &= €830\end{aligned}$$

The Retailer's margin for this product per liter is currently €0.83 (to the base of 880 liters of volume). Now, with the change in price and volume, the new margin can be calculated:

$$\begin{aligned}\text{New Retailer's Margin} &= \text{New Sales Value - New NSV} \\ &= €2,740 - €1,936 \\ &= €804\end{aligned}$$

The Retailer's margin for this product per liter increases to €0.91 (to the base of 880 liters volume) with a change in volume driven by change in price.

Appendix B

FrieslandCampina Margin Computational Model

A detailed view of all the variables involved for computing margins can be viewed in the following illustration. While there are larger number of variables involved at even lower levels, we created this model to provide a higher-level understanding of parameters and map their dependencies.

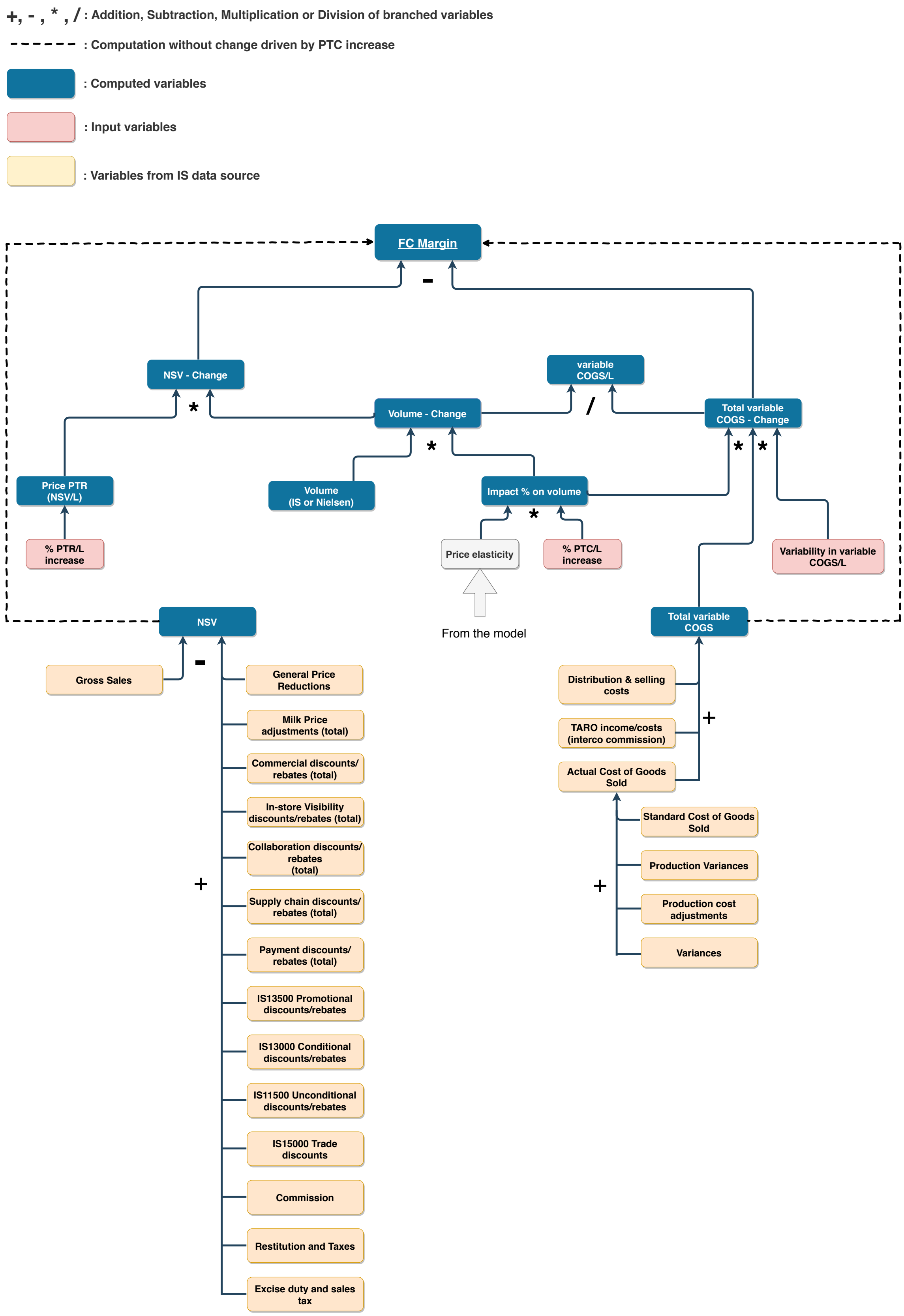


Figure B.1: FrieslandCampina Margin variables

Appendix C

Notation and Terminologies for User Visualization Tasks

The dimensions of the design space discussed by the authors in [18], relates to 4 questions; WHY, HOW, WHAT and WHERE. For these four questions, the authors describe five dimensions of goal, means, characteristics, target, and cardinality. We discuss these from their study below:

C.1 Design space dimensions

Here, we explain the terminologies used for describing the user tasks.

C.1.1 Goal

The **goal** of a visualization task defines the aim or objective with which the task is being pursued. This dimension specifies the motive of a task's action. The goal is termed as an *Exploratory analysis* when the aim is to derive hypotheses from an unknown dataset. When the user's search is undirected, the visualization task is said to be exploratory in nature. The goal can be called a *Confirmatory analysis* when the aim is to test found or assumed hypotheses from an unknown dataset. When the user's search is directed, the visualization task is said to be confirmatory in nature.

C.1.2 Means

The **means** by which a visualization task is pursued determines the method for reaching the goal. This dimension defines the action itself. In principle, it is difficult to present a definite or complete list of means to achieve a task as there could be many ways of achieving a goal. Let us describe the means we have presented in this report, in section 5.3 of chapter 5.

To *summarize* the data means to present its overview. For visualization tasks concerning analysis, the user has to *search* for elements of interest within the visualization. The *search* can be classified into more alternatives.
Lookup refers to a type of search where the users already know both what they are looking for and where it exists in the design space.
Browse is a kind of search where the users do not necessarily know the search target or what they are looking for but, they do know the location where they can look for it.
(Re-)organization may also be a means of conducting a task which includes adjusting the data to be shown by reducing or enriching it. One of the means of achieving data reduction is by extracting data by *filtering* data. On the other hand, common means of achieving enrichment are *deriving* additional data elements based on existing data elements. Deriving new data could either be a choice made by the visualization designer or the choice cold also be driven by a user of the

visual tool.
A visualization task may also be pursued with a motive to *compare* or *relate* visuals for seeking relations or differences in them.

C.1.3 Characteristics

The characteristics of a visualization task capture the aspects or facets of data that the visualization task aims to reveal. The *low-level data characteristics* are simple observations about the data such *data values* or a *data object*. The *high-level data characteristics* are more specific and complex patterns in the data, such as *trends*, *distribution*, outliers, etc. The characteristics of a visualization task could also be referred to as *features* of the data, in general without any classification.

C.1.4 Target

The target of a visualization determines which part of the data it is going to be carried out. The names of the concerned attribute values for each task, have been stated in the report to make the target explicit.

C.1.5 Cardinality

This dimension of the visualization task specifies the number of instances of the chosen target to be considered by the task. Precisely, the cardinality dimension makes a distinction whether only an individual instance or, multiple instances, or all instances of the target are to be investigated. This is a dimension of the WHERE question that has been singled out by the authors of the study under discussion. A *single* instance can be used for highlighting details, while *multiple* instances can be used for putting data into context. *All* instances of the chosen target can be considered by a task for getting a complete overview.

C.2 Notations for design space as a whole

The notation used for representing all our user tasks along these five dimensions is also motivated from the concept of using design dimensions in conjunction to form a design space, as presented in the research paper. Each individual task has been represented as a 5-tuple (goal, means, characteristics, target, cardinality) and can be used to represent a singular point in the design space.

Appendix D

Design Principles for Visual Comparison of Variables

In figure D.1, six possible designs have been illustrated to occupy the right section of the Analysis page's design space. Here, the four KPI's are to be visualized for a deep-dive analysis on how price change influences their values. The aim is to provide with design choices that enable direct comparison of variables against each other.

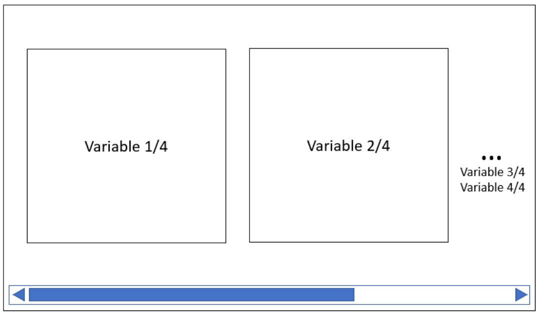
In sub-figure D.1a, we see a very basic method of scrolling the parent component container horizontally to view the four children containers containing four blocks (one for plotting each KPI). This is a very naive way of representing graphs because of the inability to see, for example, the first and the last visual graphs at the same time. This inhibits user-initiated comparison of any two variables.

In sub-figure D.1b, the user can choose two variables for direct comparison with the help of a drop-down menu. The juxtaposition of the two plots can enable visual comparison in an effective way [6].

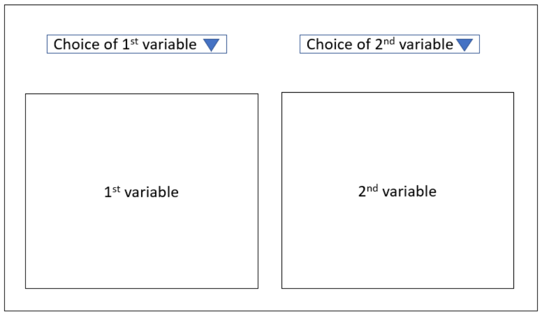
The idea portrayed in sub-figures D.1c and D.1d are slightly different from D.1b. Here, any one of the four variable's visual representation could be fixed on the left. To the right of this fixed graph, the second variable can either be chosen from the drop-down menu or switched to, with the help of three tabs, each corresponding to the three variables (except the fixed one). The limitation of these three layouts shown in D.1b, D.1c and D.1d is that the user is restricted to selecting, visualizing and comparing only two variables at a time. Apart from this, the process of selecting a variable for comparison in these design options increase the number of user clicks. Specifically, in cases where the user does not know upfront what he is looking for or wants to compare, these design layouts can make the process or duration of analysis, longer and complicated. Besides, it tests the user's memory of remembering views while he keeps switching between different (drop-down or tabs) options.

Sub-figure D.1e tries to overcome the previously stated limitation. By retaining the idea of fixing one variable, this comparison technique puts forth the possibility of viewing and comparing the other three variables at the same time. This is derived from the concept of using small multiples[4] in a context like ours where all variables are to be visualized over the same x-axis, which is the time axis. One downside of this technique is that the sizes and proportions of the graphs to compared are not equal to facilitate ease in direct comparison using juxtaposition.

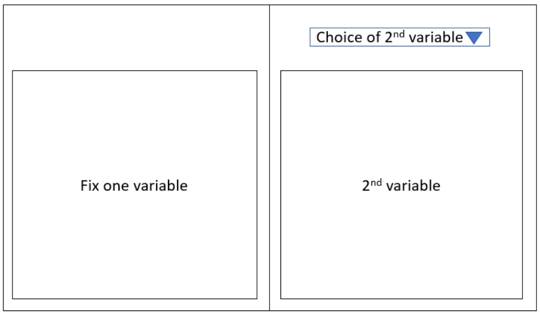
The design idea showcased in sub-figure D.1f overcomes all the above limitations. By fixing one variable on the left, the small multiples stacked vertically on top of each other, provides an overview of the three variables at the same time. This design could be further enhanced by allow-



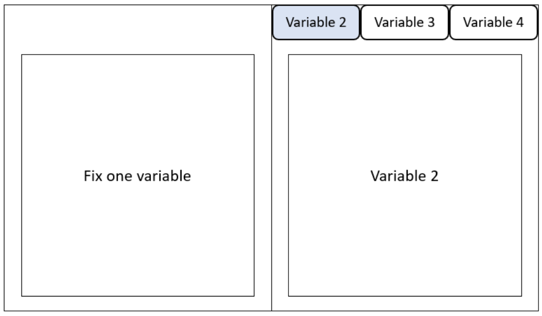
(a) Scroll horizontally to view all variables.



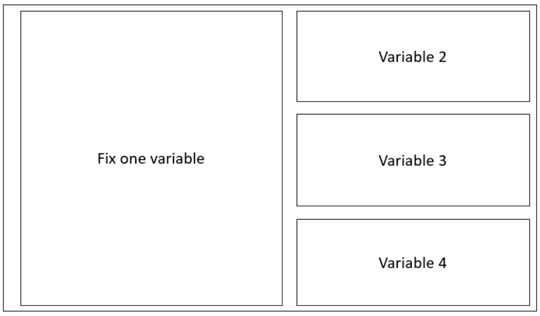
(b) Using drop-down menu to select two variables.



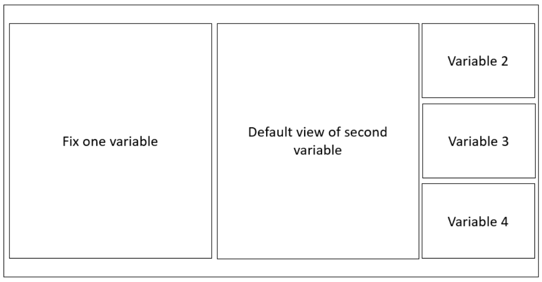
(c) Fixing one variable's view, selecting second using drop-down.



(d) Fixing one variable's view, using tabs to switch between other views.



(e) Fixing all four views. One large and three small views.



(f) Fixing first variable. Setting a second default view. Small multiples on the right with "click to expand" interactivity.

Figure D.1: Six possible layouts to position the four KPI variables with possibilities to enable variable selection for comparison.

ing the user to click on any of these small multiples for an expanded view against the first variable in the space provided to the right of it, i.e. in the middle panel that we refer to as "Default view of second variable" in the image. Until the user selects a variable for an expanded view, a default graph for one of the the variables to be visualized as small multiples, can be provided in this middle panel to fill up the empty space. This design concept seems to be the most appropriate in terms of allowing comparison of variables within the same scenario, i.e. for comparing all variables for the same simulated price settings. It also incorporates user interaction and does not restrict their choices of comparison.