BACHELOR

Diet Optimization using Linear Programming based on Schijf van Vijf

Kalayci, M.H. (Hakan)

Award date:
2022

Diet Optimization using Linear Programming based on Schijf van Vijf

EINDHOVEN UNIVERSITY OF TECHNOLOGY

Date
19 June 2022

Version 1.1

Wiskunde en Informatica
Data Science

m.h.kalayci@student.tue.nl

www.tue.nl

Author:
M.H. Kalayci (1018867)

Supervisor
Joris Wagenaar (Tilburg University)

Contents

1 Preface

This report is part of my Bachelor End Project : Schijf van Vijf. The Schijf van Vijf is a concept introduced in 1953 by the Voorlichtingsbureau voor de Voeding, which is part of the voedingscentrum. Its purpose was to advise people about what and how much to eat of different food groups. Me and 3 other students have been given the opportunity to research the area of nutritional science. I chose this project because it resonates with my trade-off with food. I'm a person who loves to eat different kinds of food, or try new tastes. However, I'm aware that it's important to know what you eat, and consume consciously. In particular I find it important that one knows his/her limits, in terms of macronutrient needs. This is part of the main topic my research will cover. Later on in this report, I will go into more detail about the context and scope of my thesis.

2 Abstract

Started as a concept in 1953, the Schijf van Vijf evolved over time into a highly advanced food advisory system. The purpose of the Schijf van Vijf is to advise people on what and how much to eat. Scientists have been trying to optimize diets. The "diet problem", created by Stigler in the 1940's, was the first mathematically proposed optimization problem for diets. This problem has been tackled by various researchers, to the point where high-performing computers were applied to solve such problems. In our research, we have applied Linear Programming techniques, while quantifying information and requirements from the Schijf van Vijf. A model has been created, which is fed with data, parameters and constraints, to output an optimized diet. The model is analyzed and an experiment is run to see how it behaves for different types of inputs. We have come to the conclusion that the model outputs a better balanced diet for certain inputs. By researching this area within nutritional science, we hope to shine some light on the subject matter, and inspire other researchers to build upon the proposed model.

3 Introduction

3.1 The project

The Schijf van Vijf was introduced to the Netherlands in 1953, after the second world war. Its main purpose then was to advise people what and how much to eat. The first version of the Schijf van Vijf was based on the “wheel of good eating”, a concept introduced earlier in the United States. The Schijf van Vijf has been changed and/or updated over time. The last drastic change has been in 2016, where the logo, advice, information for consumers and professionals, as well as the tools available have been updated. In this project, we have been given the opportunity to use our creativity to propose and research relevant research questions.

3.2 Problem Statement

Obesity and other noncommunicable illnesses are becoming more common in high-income countries. In low-income populations, obesity and undernutrition coexist. To be more precise, around 2 billion people are overweight or obese, 2 billion have nutritional deficiencies, and approximately 800 million are still famished as a result of poverty and underdeveloped food systems.[2]

The existing food production system in more economically developed countries entails a large environmental burden, which is exacerbated by the increase of industrial and agricultural output in rapidly emerging countries such as India and China.

Given this background, the notion of sustainable diets, that is, diets that are “protectful and respectful of biodiversity and the environment, culturally acceptable, accessible, economically fair and affordable, nutritionally adequate, safe and healthy, while optimizing natural and human resources”[3] become more relevant, in both low and high income countries. The pursuit of these sustainable diets has made researchers investigate and model this problem. In the context of this thesis, we will be focussing on mathematical diet optimization.

Mathematical diet optimization, also known as diet modeling or diet optimization, was pioneered in the 1940’s by Georges Stigler, who used diet as an example to transform a difficult issue into a mathematical model known as “the diet problem”. The purpose of the diet problem was to identify a set of foods that meet daily nutritional needs at the lowest possible cost. Dantzig created a technical solution for this problem using the Simplex method, which is the foundation of linear programming.

Nowadays, optimization problems might take different forms other than linear functions, such as quadratic programming, mixed-integer programming, or goal-programming to address issues with numerous objectives.

3.3 Scope

In the scope of this project, we will investigate the effect of certain sports goals on the mathematically proposed diet. Sports and exercise are important aspects in people's lives. Sports and nutrition also go hand in hand when it comes to being healthy, and reaching certain predetermined goals. Sports Nutrition can be defined as the application of nutrition knowledge to a practical daily eating plan providing the fuel for physical activity, facilitating the repair and building process following hard physical work and achieving athletic performance in competitive events, while also promoting overall health and wellness.[4]

The sports goals that will be analyzed are bulking, cutting, maintaining/losing/gaining weight, and losing excess fat. Each of these goals require a different ratio in macronutrients. As an example, bulking, which is a combination of building muscle and gaining weight, will have higher protein and caloric requirements, in comparison to losing weight, which in contrast requires a low amount of carbs and calories. Further on in this thesis, all 5 goals and their ratios will be given.

3.4 Research Questions

This thesis will address 3 main research questions. Each research question has his own subquestions. The first research question, and its sub questions, are directly related to the Schijf van Vijf. As this project is a Schijf van Vijf project, we need to keep in line with the guidelines and advice offered by the Schijf van Vijf.[5, 6] These guidelines are analyzed, and the requirements for a diet have been derived from them. The requirements will then be implemented in Python, and in our mathematical model.

1. **<u>How can the Schijf van Vijf be quantified?</u>**
    - What are the requirements advised by the Schijf van Vijf?
    - How can the requirements of the Schijf van Vijf be incorporated in a mathematical model?

The next question is related to the daily nutritional requirements of an individual. We must know the formulas to calculate them, such that we can further our research. Next to this, we will also see how these requirements change when we are focussed on a different sports goal. Sports goals include bulking, cutting, maintaining/losing/gaining weight, and fat loss.

2. **<u>What are the optimal nutritional requirements for an individual?</u>**
    - What are the essential nutritional requirements?
    - How can we calculate these nutritional requirements in the context of one individual?
    - How do the values of the requirements change when certain sports goals are implemented?

The third and most thorough question is concerned about the mathematical model which is used to calculate a diet, based on an individual's calculated daily nutritional needs. This model is further explained in the methodology part of this thesis.

3. **<u>What is the optimal diet for an individual?</u>**
    - How does the optimal diet change, in regards to different sports goals?

Aforementioned research questions (and sub questions), determine the scope of our research. The focus will be on finding the optimal diet based on any kind of preset sports requirements. The research will include an adequate analysis about these requirements, and its findings will be presented later on. Furthermore, it is essential that we do not stray too far away from the Schijf van Vijf. Therefore, the guidelines offered by the Voedingscentrum are acknowledged as the base for our mathematical model and analysis.

3.5 Scientific Relevance

As previously mentioned, diet optimization was pioneered by George Stigler in the 1940s. Stigler chose diet as an example to translate a complex problem into a mathematical model called "the diet problem".[7] The Stigler "Diet Problem" is a typical question of resource optimization or, in mathematical terms, of minimization of a linear function subject to multiple linear constraints, also called linear programming".[8] Later on, in around 1947, Dantzig came up with a technical solution to this problem. The method Dantzig developed was called the Simplex algorithm. Dantzig tested his model on his own diet. He attempted to reduce his caloric intake to 1500 calories, and programmed an objective function to maximize the feeling of being full. The solution he found was rather extraordinary, as his method proposed a diet of 200 bouillon cubes per day. These results led to the first upper limits being added to Linear Programming.[9]

A few years later, Dantzig presented a linear program and began utilizing an IBM 701 computer in the early 1950s.[9] The development of a solution to diets was heavily reliant on the development of computers with high calculation capacity. The difficult computations required for LP were only practically achievable when fast computer technology became accessible.

Soden and Fletcher from the University of Salford in the UK, were also among LP pioneers. Their LP concepts were clear in the 90s, and are still being used nowadays. They used an objective function based on individual dietary preferences. In partnership with professional dieticians, they created the "Microdiet System, 1990", a computer program which was deployed in several top UK hospitals. They presented a computational method for constructing individually acceptable diets by modifying a chosen diet to meet nutritional requirements.[10] The calculation starts with the person's current dietary intake. This was altered using LP methods which use vectors to make tiny modifications to the food quantities to meet specific targets. Sequential modification was implemented to identify changes that are acceptable to the individual. The maximum capacity of this program was 100 foods and 30 constraints. Based on these findings, Fletcher et al. created a computational technique for generating individually appropriate diets using Linear Programming.[11]

4 Methodology

4.1 Relevant theory

As we have touched in the previous section, Linear Programming (LP) is a strategy for tackling optimization problems that requires optimizing a linear function while keeping decision variables' equality or inequality constraints in mind. An objective function plus a set of constraints in the form of a system of equations or inequalities comprise a mathematical optimization model. Before a satisfactory objective function can be constructed, the variable selection procedure must be repeated several times. Linear Programming is a type of optimization problem in which both the objective function to be optimized, and the constraints are linear in terms of the decision variables. The Simplex approach, created by Dantzig in 1948 and based on numerical linear algebra methods, is typically used to tackle LP issues.[12] Linear programming is being successfully applied to problems of design of diets, conservation of resources, economic growth prediction, transportation systems. In this paper, we propose to use an LP toolkit library in Python, namely PuLP. This library consists of functions and methods which allow us to set up and solve LP problems.[13]

A linear programming problem can be defined as a problem of optimizing (minimizing or maximizing) a linear function subject to linear constraints. A feasible vector in which the objective function gets the value is called optimal. A feasible optimal problem is said to be unbounded if the objective function can assume arbitrarily large values at feasible vectors. If this is not the case, the problem is said to be bounded. The value of a bounded feasible optimum problem is the optimum value of the objective function as the variable ranges over the constraints set. Here, optimum can either be a minimum or maximum. Any linear problem consists of 4 elements:

1. A set of decision variables
2. Parameters
3. The objective function
4. A set of constraints

To solve a linear problem, one must find an objective function that minimizes, maximizes, or achieves a specific goal, while the variables satisfy the constraints of the model.

In our case, it's similar to the classical diet problem, with minor changes. The classical diet problem has to supply the required nutrient needs at the lowest possible cost for n distinct types of foods $F_1 ... F_n$, which supplies varying quantities of the k nutrients, $n_1 ... n_k$, that are essential for good health.

We alter this problem by changing the objective function, in this case "lowest possible cost". For our analysis, calories will be regarded as our cost and therefore objective function. The reason for this is the fact that we are interested in researching the diets our LP model outputs, based on different sports goals. These sports goals all are either lower bounded, or upper bounded with calories. Therefore minimizing (or maximizing, depending on instance) the calories based on constraints, should be an appropriate objective function.

For n different types of food, $F_1 ... F_n$, nutrients $n_1 ... n_k$, let c_i be the amount of calories per unit of food F_i, let l_j be the minimum daily requirement of nutrient n_j, let u_j be the maximum daily allowance of nutrient n_j, and let a_{ij} be units of nutrient n_j contained in one unit of food F_i.

Mathematically, the problem can be stated as such: if x_i is the number of portions of food F_i, then the objective function of the diet is :

$f(x_1 ... x_n) = \sum_{n=1}^{n} c_n x_n = X^T C$ where $X = (x_1\, x_2\, x_n)^T$ and $C = (c_1\, c_2\, c_n)^T$.

The amount of nutrient contained in this menu is $n_j = \sum_{m=1}^{m} a_{mj} x_m = X^T * A$, where A is the matrix of the amount of nutrients of the different foods.

Sets	Explanation
F	**Set of distinct food types, indexed by n**
N	**Set of nutrients, indexed by k**
c_i	**Amount of calories per unit of food F_i, indexed by i**
l_j	**Minimum daily requirement of nutrient n_j, indexed by j**
u_j	**Maximum daily requirement of nutrient n_j, indexed by j**
a_{ij}	**Units of nutrient n_j contained in one unit of food F_i**
x_i	**Number of portions of Food F_i**

4.2 Research strategy

Before we can start modeling our LP problem, we must first calculate the daily needs. The daily needs are calculated as follows :

1. <u>Calculate the BMR</u>

```
def calculate_bmr():
#Input
    weightInKgs = int(input("Weight in Kgs: "))
    heightInCms = int(input("Height in Cms: "))
    age = int(input("Age: "))
    maleOrFemale = input("are you a (m)ale or (f)emale? ").lower()

#BMR Formula implemented -- Harris Benedict Equation
    if maleOrFemale == 'm':
        bmr = int((10 * weightInKgs) + (6.25 * heightInCms) - (5 * age) + 5)
    elif maleOrFemale == 'f':
        bmr = int((10 * weightInKgs) + (6.25 * heightInCms) - (5 * age) - 161)
    return bmr
```

In this code fragment, we can see how the Basal Metabolic Rate is calculated. We ask the individual to input his/her weight in kg, height in cm, age in years, and gender as m or f, defined as male or female respectively. After the input has been given, the Harris Benedict Equation [18], given as $bmr = 10 * weight + 6.25 * height - 5 * age + 5$ for males, And $bmr = 10 * weight + 6.25 * height - 5 * age - 161$ for females, has been implemented to return the correct BMR for the individual.

2. <u>Calculate the daily caloric needs based on BMR and activity level</u>

```
def calculate_daily_caloric_needs(bmr):
    print(
        """
        1 = little to no exercise, inactive job
        2 = light training (1 to 3 times a week)
        3 = average training (3 to 5 times a week)
        4 = heavy training (6 to 7 times a week, or 3-4 times intensive)
        5 = extreme training (daily intense trainings)
        """
     )
    activityLevel = int(input("Select your activity level:"))
        #ALI = Activity Level Index - hardcoded values
    if activityLevel == 1:
            ALI = 1.2
    elif activityLevel == 2:
            ALI = 1.375
    elif activityLevel == 3:
            ALI = 1.55
    elif activityLevel == 4:
            ALI = 1.725
    elif activityLevel == 5:
            ALI = 1.9
    dailyCalNeeds = int(bmr * ALI)
    #how much calories you need to maintain your current weight.
    return dailyCalNeeds
```

In the second function, we calculate the daily caloric needs based on BMR and activity level. The BMR was just calculated, so we only require an input for activity level. Activity level is a number which ranges from 1.2 up to 1.9, with a difference of 0.175 between each level. This level represents how active the individual is in terms of movement and sports. It is important to multiply this factor with our BMR, to further implement the details of an individual. This will eventually result in a better model. What we calculate by multiplying the activity level with our BMR, is the individual's total daily calorie needs, in order to stay the same weight. When we take sports goals into account, it is highly likely that someone wants to either lose or gain weight. Therefore we must adjust this number.

3. Adjust the caloric needs based on a weight loss/maintain/gain goal you set.

```
def adjust_daily_caloric_needs(dailyCalNeeds):
    #losing 1 kg of fat equals 7700 calories.
    #to lose 0.5kg fat every week, we have to reduce our dailyCalNeeds by ((7700/2)/7)=3350 =~ 478 calories
    perWeekFatLoss = int(input("how much grams of fat do you want to lose weekly? "))
    adjustedDailyCalNeeds = dailyCalNeeds - (7700/(1/(perWeekFatLoss/1000))/7)

    return adjustedDailyCalNeeds
```

In this third step, we define how many grams of weight we want to lose, or gain per week. We implement this by defining 1000 grams of fat equal to 7700 calories.[19] We divide this number by 7 , to calculate how much we must differ from the daily caloric needs calculated in step 2.

4. Calculate the macronutrient needs out of the caloric needs.

```
def calculate_macros(cals):

    cals_from_protein = int(.4 * cals) # (given 40%p/40%c/20%f ratio)
    cals_in_protein = int(cals_from_protein/4) # 1 gram of protein = 4 kcal
    cals_from_carbs = int(.4 * cals)
    cals_in_carbs = int(cals_from_carbs/4) # 1 gram of carb = 4 kcal
    cals_from_fat = int(.2 * cals)
    cals_in_fat = int(cals_from_fat / 9) # 1 gram of fat = 9 kcal

    print("Calories from Protein: " + str(cals_from_protein) + " / " + str(cals_in_protein) + " grams of protein.")
    print("Calories from Carbs: " + str(cals_from_carbs) + " / " + str(cals_in_carbs) + " grams of carbs.")
    print("Calories from fat: " + str(cals_from_fat) + " / " + str(cals_in_fat) + " grams of fat.")
```

In this next step, we calculate the ratio in which our macronutrients proteins, carbs and fat must be distributed over our daily caloric goal. In this example fragment, a 40/40/20 ratio has been used, respectively for protein, carbs and fats. In this table below, we define the ratios for the different sports goals that will be analyzed.

Goal	Protein	Carbohydrates	Fat
Bulking	**25%**	**40%**	**35%**
Cutting	**25%**	**55%**	**20%**
Weight Loss	**45%**	**35%**	**20%**
Weight Maintenance	**30%**	**45%**	**25%**
Weight Gain	**30%**	**40%**	**30%**

4.3 Data & Analysis

The Dataset used for our analysis is the NEVO-2021 dataset provided by the RIVM. In this dataset, we can find data for 2207 different types of food, and more than 100 different nutrients. Within the defined scope of this analysis, we are only interested in macronutrients, so only 4 different nutrients (calories, protein, carbs and fat) will be taken into our model.

The dataset is first cleaned, so the first step is to remove all rows which have no value for any of these 4 nutrients. Secondly, we remove all duplicate rows. We then change the column names to make it more readable and easier programmable. A fragment of the resulting dataset can be seen below.

	food_group	food_name	quantity	contains_traces_of	is_fortified_with	kcal	protein	fat	carbs
2176	Groente	Paprika oranje rauw	per 100g	NaN	NaN	28	0.8	0.3	4.7
2177	Groente	Groente soep- gekookt	per 100g	NaN	NaN	31	1.1	0.3	4.6
2178	Groente	Amsoi gekookt	per 100g	NaN	NaN	21	2.0	0.2	1.8
2179	Groente	Paksoi gekookt	per 100g	NaN	NaN	17	1.6	0.2	1.7
2180	Groente	Pastinaak rauw	per 100g	NaN	NaN	71	1.8	1.1	11.0

Afterwards, we fill NaN values in the contains_traces_of, and is_fortified_with columns. These columns are later on used to filter food types based on an individual's allergies and medical restrictions. After filling these two columns, we define portion sizes per food_group. In the table below you can see the specific portion sizes. All these portion sizes are derived from the Schijf van Vijf guidelines. Take into account here that unnecessary food groups, such as alcoholic drinks, soups, spices, sweets, cakes, baby food, sauces, and combined foods have been deleted by removing all rows which are in any of these food groups. We do this because the Schijf van Vijf explicitly states that these foods are not healthy for daily consumption.[1]

Food Group	Portion Size (g)
Fruits	**100**
Vegetables	**50**
Fats and Oils	**15**
Cheese	**20**
Milk and Dairy products	**150**
Nuts and seeds	**25**
Eggs	**50**
Meats and Poultry	**100**
Fish	**100**
Bread	**35**
Legumes	**60**

Grain Products	**45**
Potatoes and Root vegetables	**75**

Initially, the values for the macronutrients in this dataset were per 100 grams of a certain food type. Having defined the portion sizes per food group, we can calculate all nutrients per portion, instead of per 100 grams. which were the initial values in the dataset. We do this by dividing all values by 100, and multiplying them by the portion size. This results in the following dataframe:

	food_group	food_name	quantity	contains_traces_of	is_fortified_with	kcal	protein	fat	carbs	flagCol	index_col	portion_size	sv5
0	Brood	Knackebrod gem	per 100g	X	X	133.00	4.060	2.065	22.500	0	0	35.0	1
1	Brood	Brood tarwe-	per 100g	X	ID	82.60	3.430	0.595	15.015	0	1	35.0	1
2	Brood	Brood rogge- donker	per 100g	X	ID	67.55	1.960	0.455	12.460	0	2	35.0	1
3	Brood	Brood rogge- licht	per 100g	X	ID	81.55	2.870	0.525	14.980	0	3	35.0	1

Lastly, we filter the complete dataset based on whether or not they are in the Schijf van Vijf or not. This has been manually checked using the “Is it in the Schijf van Vijf” tool provided on the voedingscentrum website. This final step has been done manually, because the data available to do this was not complete enough to do automatically, and no functional explanation is given on the used tool. This step however, decreased the dataset from 2207 food types, to 613. A significant increase, which will make our model require less computation time.

Now we have transformed all the data and are ready to use it for our Linear Programming model. Our LP model has been programmed with the *pulp* library in Python. The library is imported and an LP Problem has been created. Secondly, the decision variables are created, including subsets of them for more advanced constraints. The decision variables are all integer variables, because we want our model to output portions, and not a continuous output (grams). Next up, the objective function is defined as “total cost of the balanced diet, in calories”. As a programming function, it can be written as such.

```
#objective function
prob1 += lpSum([costs[i]*food_vars[i] for i in food_indexes]), "Total cost(kcals) of the balanced diet"
```

After defining our objective function, we must add linear constraints to the model. A table of all the linear constraints which are implemented is given below. These constraints are derived from the Schijf van Vijf guidelines.

Constraint	Value
Maximum total of kcal	**1.00 * Calculated Kcal**
Minimum total of carbs (g)	**0.95 *Calculated carbs**
Maximum total of carbs (g)	**1.00 * Calculated carbs**
Minimum total of protein (g)	**0.95 * Calculated protein**
Maximum total of protein (g)	**1.00 * Calculated protein**
Minimum total of fat (g)	**0.95 * Calculated fat**
Maximum total of fat (g)	**1.00 * Calculated fat**
Minimum 200 grams of fruit	**(2x100g)**

Maximum 300 grams of fruit	(3x100g)
Minimum 250 grams of vegetables	(5x50g)
Maximum 300 grams of vegetables	(6x50g)
Minimum 175 grams of bread	(5x35g)
Maximum 245 grams of bread	(7x35g)
Minimum 40 grams of cheese	(2x20g)
Maximum 60 grams of cheese	(3x20g)
Minimum 45 grams of grain products	(1x45g)
Minimum 75 grams of potatoes/root vegetables	(1x75g)
Minimum 3 portions of Grain and potatoes/root vegetables combined.	X1 + X2 > 3
Maximum 4 portions of Grain and potatoes/root vegetables combined.	X1 + X2 < 4
Minimum 300 grams of dairy products	(2x150g)
Maximum 450 grams of dairy products	(3x150g)
Minimum 30 grams of fats & oils	(2x15g)
Maximum 45 grams of fats & oils	(3x15g)
Minimum 25 grams of nuts & seeds	(1x25g)
Maximum 25 grams of nuts & seeds	(1x25g)
Maximum 1 portion of legumes	(1x60g)
Maximum 2 portions of legumes, meats and fish combined	X1+ X2+X3 < 2

All given linear constraints are implemented in the LP model. At this point, it is possible to run the computation and find a solution which fits all the constraints. Recall the sub research question : “How does the optimal diet change, in regards to different sports goals?”. To answer this question, the following analysis is performed.

To analyze the differences in sports goals within our model, we must perform the calculations for each of these goals, while using a fixed individual’s parameters for all of the calculations. We compute the model 5 times for each sports goal, for the same individual. The individual’s parameters for the 5 different calculations are given in the figure below:

Parameter	Computation 1	Computation 2	Computation 3	Computation 4	Computation 5
Sports goal	Bulking	Cutting	Weight-Loss	Weight-Maintain	Weight-Gain
Weight(in kg)	90	90	90	90	90
Height (in cm)	180	180	180	180	180
Age	24	24	24	24	24
Gender	Male	Male	Male	Male	Male
Activity level	2	2	2	2	2
Weight-change per week(g)	500	-500	-500	0	500
Protein/carb/fat ratio	25/40/35	25/55/20	45/35/20	30/45/25	30/40/30

Allergies	**None**	**None**	**None**	**None**	**None**
Restrictions	**None**	**None**	**None**	**None**	**None**

The results of these computations are presented and discussed in the Results section.

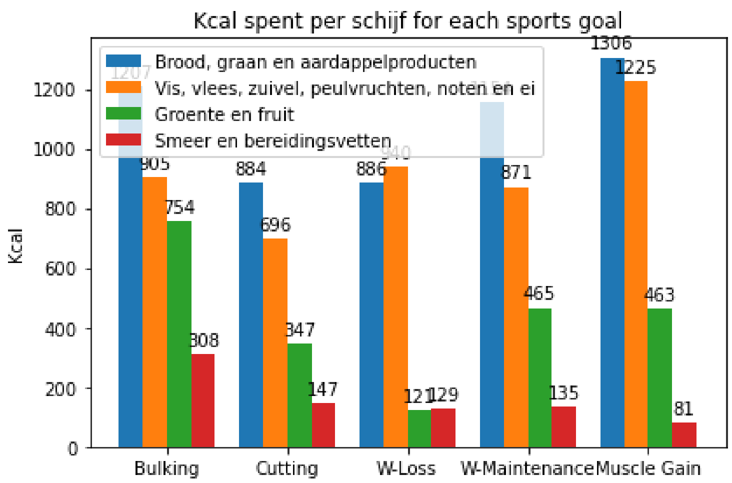
5 Results

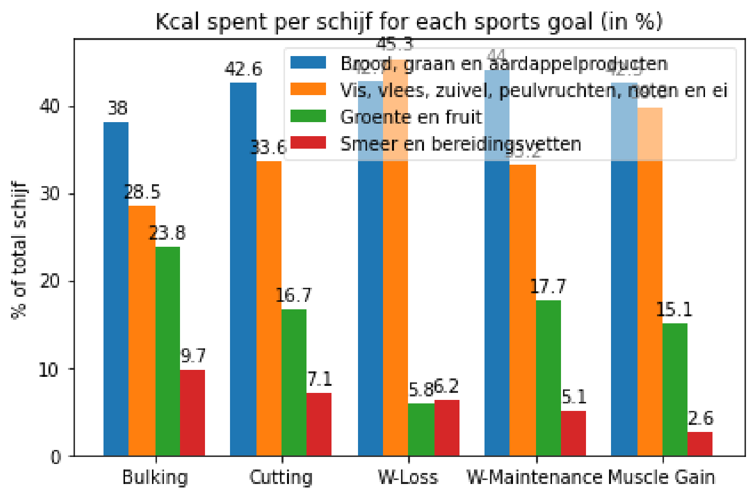
As stated before, the model has been computed 5 times. A fragment of the output can be seen in the image below. In this table, we see how many portions of each product we must eat in order to fulfill all implemented constraints. The values for *kcal* are summed per slice of the Schijf van Vijf. These values are presented in a bar chart.

	index_col	product_value	food_group	food_name	quantity	contains_traces_of	is_fortified_with	kcal	protein	fat	carbs
0		2	Vetten en oliën	Halvarineproduct Albert Heijn light		X	CHOCAL, FOL, FOLAC, RETOL, RIBF, THIA, VITA_RA...	81.00	0.00	9.000	0.00
1		2	Fruit	Citroen		X	X	72.00	1.60	0.600	6.00
2		1	Fruit	Pruimen gedroogd		X	X	220.00	2.00	0.000	45.00
3		1	Peulvruchten	Bonen soja-gedroogd		VITC	X	250.00	21.54	11.160	9.40
4		1	Vlees en gevogelte	Kalfssukadelappen bereid		X	X	158.00	27.00	5.500	0.00
5		1	Melk en melkproducten	Yoghurt 0% vet Activia m vruchten		X	X	79.50	6.90	0.150	11.25
6		2	Melk en melkproducten	Skyr naturel magere		X	X	100.00	31.00	0.000	11.10

5.1 Analysis results

The results of our analysis have been visualized and plotted in a bar chart. In this chart, we can see significant differences between the models output, for the different sports goals. We can also clearly see how the model has "spent" its "calorie budget". We have neglected the "Dranken" schijf, which includes water coffee and tea for this specific analysis.





5.2 Findings

Some interesting findings we can derive from this chart:

- *The protein/carbs/fat ratio is crucial in getting optimal results from the model. We can see that only a slight 5% adjustment in the ratios can result in a significantly different output.*
- *For bulking, there is a significant amount of calories spent on fruits and vegetables. This occurs because the constraints are limited, and the only way to keep within bounds is to use calorie-dense fruits.*
- *For weight loss, the calories spent on fruits and vegetables is significantly low. This means that the model specifically includes calorie efficient fruits, in order to fulfill all requirements.*
- *The most balanced diets seem to be the output of "Cutting" and "Weight-Maintenance".*

5.3 Research questions answered

Having performed our analysis, we can touch back on the research questions which guided our research, and answer them with the knowledge we have gained.

First question was regarding the quantification of the Schijf van Vijf. The quantification can be done by deriving all lower bounds and upper bounds for certain food types, and nutrients, from the Richtlijnen document provided by the Voedingscentrum. All these lower and upper bounds have been incorporated in our model through linear constraints. Other parameters such as the protein/carb/fat ratios and daily nutritional requirements, are implemented in an earlier stage of our program, instead of as a linear constraint.

The second research question was concerned with the optimal nutritional requirements for an individual. The optimal nutritional requirements for an individual is determined by calculating the BMR and multiplying it by the ALI (Activity Level Index). The outcome is the number of calories one should eat to maintain his/her current weight. To calculate the BMR and ALI, it is required that the individual input his/her age, height, weight, gender and activity level. For macronutrient requirements, the required values significantly depend on the individual's goals. E.g. For a weight loss diet, lower carbs and higher protein is suggested.

Lastly, the third research question concerning the optimal diet, and how the resulting optimal diet behaves with respect to different sports goals, can be answered. In the findings presented in the previous section, we have summed up the main insights gathered from this analysis.

6 Conclusion

6.1 Discussion

Reflecting on this paper and its analysis, there exist some points which are open to discussion. In order to clarify these points, justification and elaboration will be provided in this section. Firstly, it has been decided to investigate the matter on a daily basis, instead of weekly. The Schijf van Vijf allows certain foods to be eaten 2 or 3 times per week (week choices), but if you look at the problem on a daily basis, these foods are not allowed. Therefore, we have chosen to optimize a daily diet.

The second discussion point is portion sizes. It would be ideal to have a portion size per food type, to minimize discrepancies. However, seeing as it would be very time-consuming to implement this, we have opted for a solution to assign each food group a fixed portion size. These fixed portion sizes have been given in a table in the methodology section of this thesis.

Thirdly, we can discuss the lower and upper limits for our linear constraints. We have decided to use the calculated macronutrient requirement as our upper bound, and 95% of this value as our lower bound. Our LP model is a LP-Maximize model, meaning that it will try to get the highest value within each constraint. Therefore it makes sense to use the requirement as our upper bound.

Another important point of discussion is the process of filtering our dataset based on whether or not it is in the Schijf van Vijf. As stated before, the tool on the voedingscentrum website has been used (manually), to check whether a type of food is in the Schijf van vijf or not. This resulted in ensmallening the dataset from about 2100 rows to 600. This was an important step in achieving our result, as foods who are not optimal have been filtered out. Also, reducing the size of the dataset made our computations run faster.

The last discussion point we're going to cover are the ratios for macronutrients. The ratios are not exclusively for its specific sports goal. Namely, it can differ among individuals. In general, these macronutrient ratios are used by dieticians and professional sporters.

6.2 Recommendations and further research

Concluding, the results gained from this research and analysis have been insightful. Even though the findings are not disruptively impactful, it still allows room for some interpretation. For researchers who would like to build upon this model and analysis, it is recommended to find good, consistent data to work with. What can also be proposed, is to fine-tune the proposed model to work with different kinds of food and recipes. Another possible addition to this research might be to include micronutrients, such as vitamins and fibers, and analyze how the model behaves when these are included.

7 Literature

7.1 List of references

1. Richtlijnen Schijf van Vijf. Voedingscentrum 2019.
2. Food and Agriculture Organization of the United Nations. International Fund for Agricultural Development. the United Nations Children's Fund. World Food Programme. WHO . World Health Organization; Geneva: 2018. The state of food security and nutrition in the world.
3. Proceedings of the International Scientific Symposium BIODIVERSITY AND SUSTAINABLE DIETS UNITED AGAINST HUNGER 3–5 November 2010 FAO Headquarters, Rome
4. Brink, Elizabeth et al. Development of healthy and sustainable food-based dietary guidelines for the Netherlands. The Netherlands Nutrition Centre (Voedingscentrum), The Netherlands: National Institute for Public Health and the Environment (RIVM), Bilthoven, The Netherlands[2018]
5. Richtlijnen Schijf van Vijf. Voedingscentrum. [First publication 2016]
6. Stigler GJ. The cost of subsistence. J Farm Econ 1945;27:303.
7. Buttriss JL, Briend A, Darmon N, Ferguson EL, Maillot M, Lluch A. Diet modelling: how it can inform the development of dietary recommendations and public health policy. *Nutr Bull.* (2014) 39:115–25. doi: 10.1111/nbu.12076
8. Dantzig G. The diet problem. Interfaces (Providence) 1990;20:43–7
9. Soden PM, Fletcher LR. Modifying diets to satisfy nutritional requirements using linear programming. *Br J Nutr.* (1992) 68:565–72. doi: 10.1079/BJN19920115
10. Fletcher LR, Soden PM, Zinober ASI. Linear programming techniques for the construction of palatable human diets. *J Op Res Soc.* (1994) 45:489–96. doi: 10.1057/jors.1994.76
11. Hussain, Mohammad & Hussain, Mohammad. (2019). Simplex method to Optimize Mathematical manipulation.
12. Mitchell, Stuart et al. PuLP: A Linear Programming Toolkit for Python. (2011)
13. Frankenfield DC, Muth ER, Rowe WA. The Harris-Benedict studies of human basal metabolism: history and limitations. J Am Diet Assoc. 1998 Apr;98(4):439-45. doi: 10.1016/S0002-8223(98)00100-X. PMID: 9550168.
14. Thomas, D. M., Gonzalez, M. C., Pereira, A. Z., Redman, L. M., & Heymsfield, S. B. (2014). Time to correctly predict the amount of weight loss with dieting. *Journal of the Academy of Nutrition and Dietetics*, *114*(6), 857–861. https://doi.org/10.1016/j.jand.2014.02.003