

CMSC 150 PYTHON PROJECT

An app that has a diet solver as its main features and generic solver as its additional feature. The diet solver involves the use of Simplex Method in minimization with mixed constraints. The generic solver on the other hand, has a feature to generate a polynomial regression function based on given data points and another feature that involves quadratic spline interpolation.

Diet Solver

The objective of this feature is to identify the most cost-effective and nutritious combination of foods that will fulfill all daily nutritional requirements. The combination of foods will be based upon the food options selected by the user. This problem is formulated as a linear program with the objective of minimizing cost under specified constraints and ensuring nutritional adequacy. These constraints control factors such as number of calories and amounts of vitamins, minerals, fats, sodium and cholesterol in the diet. Additionally, each food option is restricted to a range of 0-10 servings. The program employs simplex method set up as a dual problem to solve for the optimal combination of foods.

Polynomial Regression

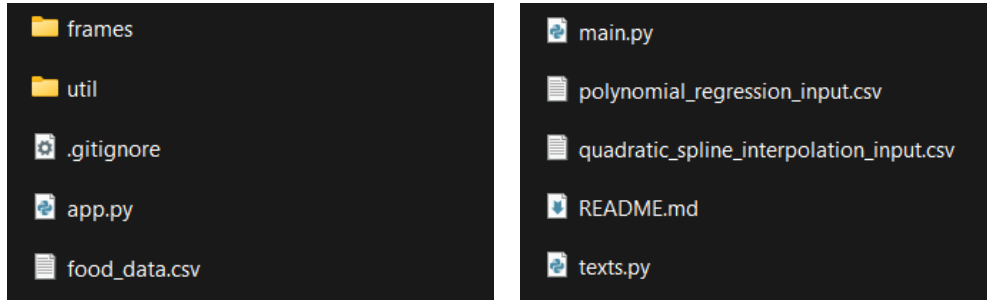
Polynomial regression is a statistical method used to model the relationship between a dependent variable and one or more independent variables. This calculator employs polynomial regression to generate precise polynomial equations based on user-provided data points. Users can input their datasets and specify the desired polynomial degree, allowing the calculator to determine the optimal polynomial equation that best fits the given data. The result is an accurate polynomial equation that can be utilized for predictive purposes. This tool is invaluable for tasks where a more complex relationship between variables needs to be captured, offering a higher degree of accuracy in modeling.

Quadratic Spline Interpolation

Polynomial regression is a statistical method used to model the relationship between a dependent variable and one or more independent variables. This calculator employs polynomial regression to generate precise polynomial equations based on user-provided data points. Users can input their datasets and specify the desired polynomial degree, allowing the calculator to determine the optimal polynomial equation that best fits the given data. This tool is invaluable for tasks where a more complex relationship between variables needs to be captured, offering a higher degree of accuracy in modeling. Quadratic spline interpolation is particularly useful when dealing with real-world data that may exhibit non-linear patterns within specific ranges.

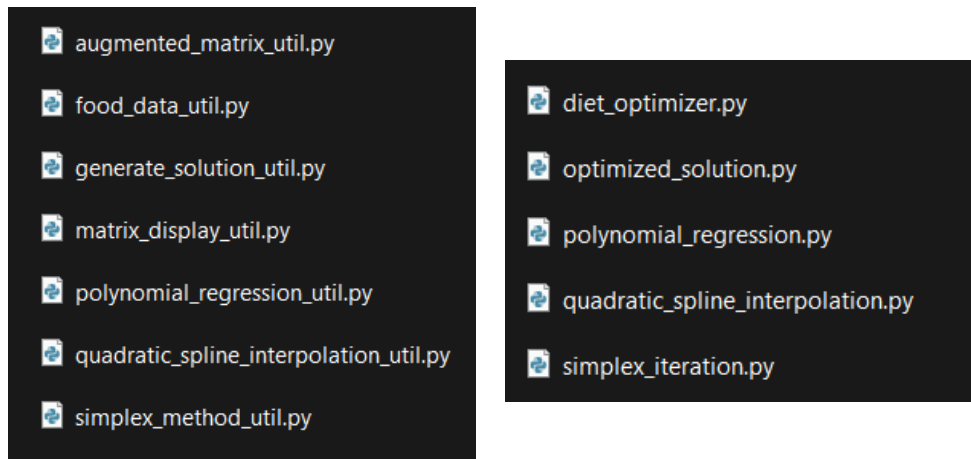
Downloading Resources

Upon downloading all resources for the project, it should contain all the following files:



Utils and Frames Folders

The utils folders contains all the logic and necessary functions for various methods and return the necessary outputs for the user to see through an interface. The frames folders on the other hand contains all the UI for the program. It contains different pages and also handles the error prompts and calls the corresponding util function needed.



Installing External Modules

Make sure you have Python and pip installed on your system before running these commands.

```
pip install tk
pip install ttkbootstrap
pip install numpy
```

If you're using a virtual environment, activate it before running the commands.

```
# Create a virtual environment
python -m venv myenv

# Activate the virtual environment (Windows)
myenv\Scripts\activate

# Activate the virtual environment (Unix or MacOS)
source myenv/bin/activate
```

Running the Program

Once you are in the correct directory, run your Python script using the python command followed by the script's filename. If you are using Python 3, you might need to use python3 instead:

```
# Navigate through file directory
cd path/to/your/script/directory

# Run the main python application
python main.py
```

Main Menu

Upon running the main application, we are greeted with the app overview and its available functions such as a diet solver, polynomial regression calculator and quadratic spline interpolation. Clicking the corresponding buttons will send the user to corresponding solver.

SOLVER TOOLKIT

An app featuring a diet solver using the simplex method as dual problem for minimization. It also includes a generic solver for generating polynomial regression functions and quadratic spline interpolation.

Diet Optimizer

The diet optimizer identifies a cost-effective and nutritious combination of foods, formulated as a linear program minimizing cost under specified constraints, utilizing simplex method as dual problem.

Polynomial Regression

The polynomial regression calculator generates accurate polynomial regression functions from given data points. Users can input datasets, specify polynomial degree, and receive an optimal polynomial equation for precise modeling.

Quadratic Spline Interpolation

The quadratic spline interpolation calculator interpolates data points using quadratic functions for each interval. It ensures accurate curve fitting within the given range, providing seamless interpolation for the provided data.

Diet Optimizer

Clicking the Diet Optimizer button will send user to the diet solver where they have the option to select foods. The program has a built-in search function to make selection easier in addition to the default checkboxes.

Diet Optimizer

Use the food search bar function or manually tick checkboxes to select desired foods.

Search desired foods

Add

- | | | | | |
|---|---|---|---|---|
| <input type="checkbox"/> Frozen Broccoli | <input type="checkbox"/> Grapes | <input type="checkbox"/> Skim Milk | <input type="checkbox"/> Oatmeal | <input type="checkbox"/> Popcorn,Air- Popped |
| <input type="checkbox"/> Carrots,Raw | <input type="checkbox"/> Kiwifruit,Raw, Fresh | <input type="checkbox"/> Poached Eggs | <input type="checkbox"/> Malt-O- Meal,Choc | <input type="checkbox"/> Potato Chips,Bbqflvr |
| <input type="checkbox"/> Celery, Raw | <input type="checkbox"/> Oranges | <input type="checkbox"/> Scrambled Eggs | <input type="checkbox"/> Pizza W/Pepperoni | <input type="checkbox"/> Pretzels |
| <input type="checkbox"/> Frozen Corn | <input type="checkbox"/> Bagels | <input type="checkbox"/> Bologna,Turkey | <input type="checkbox"/> Taco | <input type="checkbox"/> Tortilla Chip |
| <input type="checkbox"/> Lettuce,Iceberg,Raw | <input type="checkbox"/> Wheat Bread | <input type="checkbox"/> Frankfurter, Beef | <input type="checkbox"/> Hamburger W/Toppings | <input type="checkbox"/> Chicknoodl Soup |
| <input type="checkbox"/> Peppers, Sweet, Raw | <input type="checkbox"/> White Bread | <input type="checkbox"/> Ham,Sliced,Ex tra lean | <input type="checkbox"/> Hotdog, Plain | <input type="checkbox"/> Splt Pea&Hamsoup |
| <input type="checkbox"/> Potatoes, Baked | <input type="checkbox"/> Oatmeal Cookies | <input type="checkbox"/> Kielbasa,Prk | <input type="checkbox"/> Couscous | <input type="checkbox"/> Vegetbeef Soup |
| <input type="checkbox"/> Tofu | <input type="checkbox"/> Apple Pie | <input type="checkbox"/> Cap'N Crunch | <input type="checkbox"/> White Rice | <input type="checkbox"/> Neweng Clamchwd |
| <input type="checkbox"/> Roasted Chicken | <input type="checkbox"/> Chocolate Chip Cookies | <input type="checkbox"/> Cheerios | <input type="checkbox"/> Macaroni,Ckd | <input type="checkbox"/> Tomato Soup |
| <input type="checkbox"/> Spaghetti W/ Sauce | <input type="checkbox"/> Butter,Regular | <input type="checkbox"/> Corn Flks, Kellogg'S | <input type="checkbox"/> Peanut Butter | <input type="checkbox"/> New E Clamchwd,W/ Milk |
| <input type="checkbox"/> Tomato,Red,R ipe,Raw | <input type="checkbox"/> Cheddar Cheese | <input type="checkbox"/> Raisin Brn, Kellg'S | <input type="checkbox"/> Pork | <input type="checkbox"/> Crm Mshrm Soup,W/Milk |
| <input type="checkbox"/> Apple,Raw,W/ Skin | <input type="checkbox"/> 3.3% Fat,Whole Milk | <input type="checkbox"/> Rice Krispies | <input type="checkbox"/> Sardines in Oil | <input type="checkbox"/> Beanbacn Soup,W/Watr |
| <input type="checkbox"/> Banana | <input type="checkbox"/> 2% Lowfat Milk | <input type="checkbox"/> Special K | <input type="checkbox"/> White Tuna in Water | |

Back

See Nutrients Table

Select All

Generate

Clear Selection

CMSC 150 Numerical and Symbolic Computation

Franz Christian D. Morelos - Project Manual

Clicking the *SEE NUTRIENTS* button will show a pop-up window of all the food data with corresponding nutritional values and its price per serving.

Below is the list of foods and its corresponding cost and nutritional value per serving

Foods	Costs	Serving	Calorie	Cholesterol n	Fat g	Sodium mg	Carbs g	Fiber g	Protein g	Vit A	Vit C	Calcium mg	Iron mg
Frozen Broccoli	0.16	10.00 Oz Pkg	73.8	0.0	0.8	68.2	13.6	8.5	8.0	5867.4	160.2	159.0	2.3
Carrots,Raw	0.07	0.50 Cup Shredd	23.7	0.0	0.1	19.2	5.6	1.6	0.6	15471.0	5.1	14.9	0.3
Celery, Raw	0.04	1.00 Stalk	6.4	0.0	0.1	34.8	1.5	0.7	0.3	53.6	2.8	16.0	0.2
Frozen Corn	0.18	0.50 Cup	72.2	0.0	0.6	2.5	17.1	2.0	2.5	106.6	5.2	3.3	0.3
Lettuce,Iceberg,Raw	0.02	1.00 Leaf	2.6	0.0	0.0	1.8	0.4	0.3	0.2	66.0	0.8	3.8	0.1
Peppers, Sweet, Raw	0.53	1.00 Pepper	20.0	0.0	0.1	1.5	4.8	1.3	0.7	467.7	66.1	6.7	0.3
Potatoes, Baked	0.06	0.50 Cup	171.5	0.0	0.2	15.2	39.9	3.2	3.7	0.0	15.6	22.7	4.3
Tofu	0.31	0.25 Block	88.2	0.0	5.5	8.1	2.2	1.4	9.4	98.6	0.1	121.8	6.2
Roasted Chicken	0.84	1.00 Lb	277.4	129.9	10.8	125.6	0.0	0.0	42.2	77.4	0.0	21.9	1.8
Spaghetti W/ Sauce	0.78	1.50 Cup	358.2	0.0	12.3	1237.1	58.3	11.6	8.2	3055.2	27.9	80.2	2.3
Tomato,Red,Ripe,Raw	0.27	1.00 Tomato, 2-	25.8	0.0	0.4	11.1	5.7	1.4	1.0	766.3	23.5	6.2	0.6
Apple,Raw,W/ Skin	0.24	1.00 Fruit,3/Lb,W	81.4	0.0	0.5	0.0	21.0	3.7	0.3	73.1	7.9	9.7	0.2
Banana	0.15	1.00 Fruit,Wo/Sk	104.9	0.0	0.5	1.1	26.7	2.7	1.2	92.3	10.4	6.8	0.4
Grapes	0.32	10.00 Fruits,Wo/1	15.1	0.0	0.1	0.5	4.1	0.2	0.2	24.0	1.0	3.4	0.1
Kiwifruit,Raw, Fresh	0.49	1.00 Med Frt,Wo.	46.4	0.0	0.3	3.8	11.3	2.6	0.8	133.0	74.5	19.8	0.3

The program also has a *SELECT ALL* function if user chooses to have a wide variation of foods and for easier testing of the program's simplex method set up as a dual problem.

Diet Optimizer

Use the food search bar function or manually tick checkboxes to select desired foods.

☒ Frozen Broccoli

☒ Carrots,Raw

☒ Celery, Raw

☒ Frozen Corn

☒ Lettuce,Iceberg,Raw

☒ Peppers, Sweet, Raw

☒ Potatoes, Baked

☒ Tofu

☒ Roasted Chicken

☒ Spaghetti W/ Sauce

☒ Tomato,Red,Ripe,Raw

☒ Apple,Raw,W/ Skin

☒ Banana

☒ Grapes

☒ Kiwifruit,Raw, Fresh

☒ Oranges

☒ Bagels

☒ Wheat Bread

☒ White Bread

☒ Oatmeal Cookies

☒ Apple Pie

☒ Chocolate Chip Cookies

☒ Butter,Regular

☒ Cheddar Cheese

☒ 3.3% Fat,Whole Milk

☒ 2% Lowfat Milk

☒ Skim Milk

☒ Poached Eggs

☒ Scrambled Eggs

☒ Bologna,Turkey

☒ Frankfurter, Beef

☒ Ham,Sliced,Ex tra lean

☒ Kielbasa,Prk

☒ Cap'n Crunch

☒ Cheerios

☒ Corn Flks, Kellogg'S

☒ Raisin Brn, Kellg'S

☒ Rice Krispies

☒ Special K

☒ Oatmeal

☒ Malt-O- Meal,Choc

☒ Pizza W/Pepperoni

☒ Taco

☒ Hamburger W/Toppings

☒ Hotdog, Plain

☒ Couscous

☒ White Rice

☒ Macaroni,Ckd

☒ Peanut Butter

☒ Pork

☒ Sardines in Oil

☒ White Tuna in Water

☒ Popcorn,Air- Popped

☒ Potato Chips,Bbqflvr

☒ Pretzels

☒ Tortilla Chip

☒ Chicknoodl Soup

☒ Splt Pea&Hamsoup

☒ Vegetbeef Soup

☒ Neweng Clamchwd

☒ Tomato Soup

☒ New E Clamchwd,W/ Milk

☒ Crm Mshrm Soup,W/Mlk

☒ Beanbacn Soup,W/Watr

Upon clicking the *GENERATE* button the program will generate the optimal combination of foods that would meet the daily nutritional requirements at the cheapest cost. The *CLEAR SELECTION* enables clear all previous selection.

Optimized Food Combination

This is the most cost efficient combination of foods based on your selection that will meet your minimum nutrient daily needs.

Food	Serving	Cost
Carrots,Raw	0.24	\$ 0.02
Potatoes, Baked	3.54	\$ 0.21
Skim Milk	2.17	\$ 0.28
Peanut Butter	3.60	\$ 0.25
Popcorn,Air- Popped	4.82	\$ 0.19
Total Cost	~	\$ 0.96

Upon clicking the *GENERATE* button without a food selected proper error prompts would be display and the *SEE SIMPLEX ITERATION* button would be disable as there is not food selection to optimize

Optimized Food Combination

Given the list of selected foods, there is no way to meet the daily minimum nutrient requirements. Try choosing more foods or vary your selection.

Warning: No Food Selected

BackSee Simplex Iterations

However, upon choosing a variation of foods with feasible solution, users can choose to see each simplex iteration with its corresponding basic solution. They can select the specific iteration they want to see and generate the iteration by clicking the *GENERATE ITERATION* BUTTON

Simplex Iterations

Choose specific iteration count to display below the matrix for that iteration of the simplex method set up as a dual problem for solving the optimal food combination that would meet minimum daily nutrient requirements.

BackGenerate Iteration

5

S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20
0.0	-32.16	0.0	-111.0	0.0	9.0	1.74	0.0	-33.61	0.0	-7.21	0.0	32.16	0.0	111.06	0.0	-9.0	-1.74	0.0	33.61
0.0	-31.55	0.0	-672.7	0.0	63.18	7.94	0.0	-226.4	0.0	-43.16	0.0	31.55	0.0	672.71	0.0	-63.18	-7.94	0.0	226.4
0.0	-0.44	1.0	47.08	0.0	-6.49	-2.68	0.0	23.49	0.0	4.46	0.0	0.44	-1.0	-47.08	0.0	6.49	2.68	0.0	-23.49
0.0	-0.7	0.0	-57.89	0.0	-9.84	-1.82	0.0	33.23	0.0	8.29	0.0	0.7	0.0	57.89	0.0	9.84	1.82	0.0	-33.23
0.0	0.08	0.0	-481.0	0.0	2.86	0.68	0.0	2.0	0.0	-0.03	0.0	-0.08	0.0	481.02	0.0	-2.86	-0.68	0.0	-2.0
0.0	-0.01	0.0	-21.74	0.0	-6.64	0.35	0.0	30.2	0.0	5.99	0.0	0.01	0.0	21.74	0.0	6.64	-0.35	0.0	-30.2
0.0	-11.08	0.0	-1895.0	0.0	16.24	3.56	0.0	-49.83	0.0	-10.26	0.0	11.08	0.0	1895.5	0.0	-16.24	-3.56	0.0	49.83
0.0	-5.7	0.0	-1022.0	0.0	18.86	2.89	0.0	-68.55	0.0	-13.01	0.0	5.7	0.0	1022.6	0.0	-18.86	-2.89	0.0	68.55
0.0	-8.36	0.0	-1984.0	0.0	19.69	2.84	0.0	-73.91	0.0	-14.71	0.0	8.36	0.0	1984.4	0.0	-19.69	-2.84	0.0	73.91
0.0	-7.85	0.0	-1882.0	0.0	17.98	3.16	0.0	-60.4	0.0	-12.6	0.0	7.85	0.0	1882.8	0.0	-17.98	-3.16	0.0	60.4
0.0	-0.29	0.0	-1660.0	0.0	-3.66	-1.99	0.0	-99.74	0.0	3.89	0.0	0.29	0.0	1660.2	0.0	3.66	1.99	0.0	99.74
0.0	-19.32	0.0	-889.8	0.0	6.12	2.05	0.0	-21.85	0.0	-4.69	0.0	19.32	0.0	889.86	0.0	-6.12	-2.05	0.0	21.85
0.0	-17.93	0.0	-827.7	0.0	-6.98	1.22	0.0	30.45	0.0	5.96	0.0	17.93	0.0	827.77	0.0	6.98	-1.22	0.0	-30.45
0.0	-1.12	0.0	-897.9	0.0	0.14	1.39	0.0	-17.26	0.0	-4.27	0.0	1.12	0.0	897.96	0.0	-0.14	-1.39	0.0	17.26
250.0	290.46	0.0	1502.7	0.0	59.98	24.41	45000.0	19938.0	800.0	8.45	0.0	9.54	65.0	897.22	300.0	15.02	25.59	0.0	11.71

5	x46	x47	x48	x49	x50	x51	x52	x53	x54	x55	x56	x57	x58	x59	x60	x61	x62	x63	x64	z
0.0	0.0	0.0	3.6	0.0	0.0	0.0	0.0	4.82	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.96

This also includes proper error prompts if invalid iteration count was inputted by the user or if the user did not specify the iteration count upon clicking the *GENERATE* button

Simplex Iterations

Choose specific iteration count to display below the matrix for that iteration of the simplex method set up as a dual problem for solving the optimal food combination that would meet minimum daily nutrient requirements.

Back

Generate Iteration

Select iteration count first

Upon navigating to the polynomial regression calculator through the main menu, the user may have the option of manually typing each data points following the format specified. A *RESET* button is also included for user to easily try new estimates and datapoints.

The screenshot shows the 'Polynomial Regression' web application. It features a dark theme with white text. At the top, the title 'Polynomial Regression' is in a large, bold font. Below it, a section titled 'CSV FILE FORMAT' explains that data points are comma-separated and newlines indicate new sets of data points. A 'See Example' button is provided. The main interface includes input fields for 'Degree' (set to 3), 'Data Points X' (1,2,3,4,5), 'Data Points Y' (2,5,6,3,8,2), and 'Estimate' (5). To the right, the 'Polynomial Function' is displayed as $4.32 + -4.22857x^1 + 2.77143x^2 + -0.4x^3$, and the 'Estimate at 5.0' is shown as 2.46286. At the bottom, there are four buttons: 'Back', 'Upload CSV', 'Calculate' (highlighted with a dashed border), and 'Reset'.

If data set is large, there is also an option of uploading a csv file following the example format that can be seen by clicking the *SEE EXAMPLE* button. Upon filling in the parameter such as degree and estimate, the program will generate a polynomial regression function and the estimate at that point.

CSV File format example
This can be accessed
Upon clicking the see
Example button

CSV File Format	
x	y
01	10.00
04	25.00
06	33.50
07	15.75
08	28.00
11	34.02
15	30.00

This screenshot shows the same 'Polynomial Regression' interface as before, but with a file selection dialog box open over it. The dialog is titled 'Select CSV file for Data Points X'. It shows a file explorer view with a sidebar containing folders like 'CMSC 130', 'CMSC 150', 'MATH 10', 'YSES', and 'UP'. The main pane shows a list of files: 'util' (folder), 'food_data.csv', 'polynomial_regression_input.csv', and 'quadratic_spline_interpolation_input.csv'. The 'File name' field is empty, and the file type is set to 'CSV files (*.csv)'. 'Open' and 'Cancel' buttons are at the bottom right of the dialog.

Upon navigating to the quadratic spline interpolation page, similar to the polynomial regression page, the user may also manually type each datapoints or upload a csv file following the specified format. The program will then generate each function for each interval and estimate at a specific point, this is upon clicking the *CALCULATE* button. Ther is also a *RESET* button to make testing eaiser.

Quadratic Spline Interpolation

CSV FILE FORMAT
Each data points are comma separated and newline indicates new set of data points

[See Example](#)

Data Points X (comma-separated or CSV file):
CSV file successfully uploaded

Data Points Y (comma-separated or CSV file):
CSV file successfully uploaded

Estimate:
16

[Back](#) [Upload CSV](#) [Calculate](#) [Reset](#)

[1] Function: $0.0000x^2 + 22.7040x + 0.0000$
[2] Function: $0.8888x^2 + 4.9280x + 88.8800$
[3] Function: $-0.1356x^2 + 35.6600x + -141.6100$
[4] Function: $1.6048x^2 + -33.9560x + 554.5500$
[5] Function: $0.2089x^2 + 28.8600x + -152.1300$

Estimate at 16.0:
394.2364