Blendplot Documentation

Christopher Wells

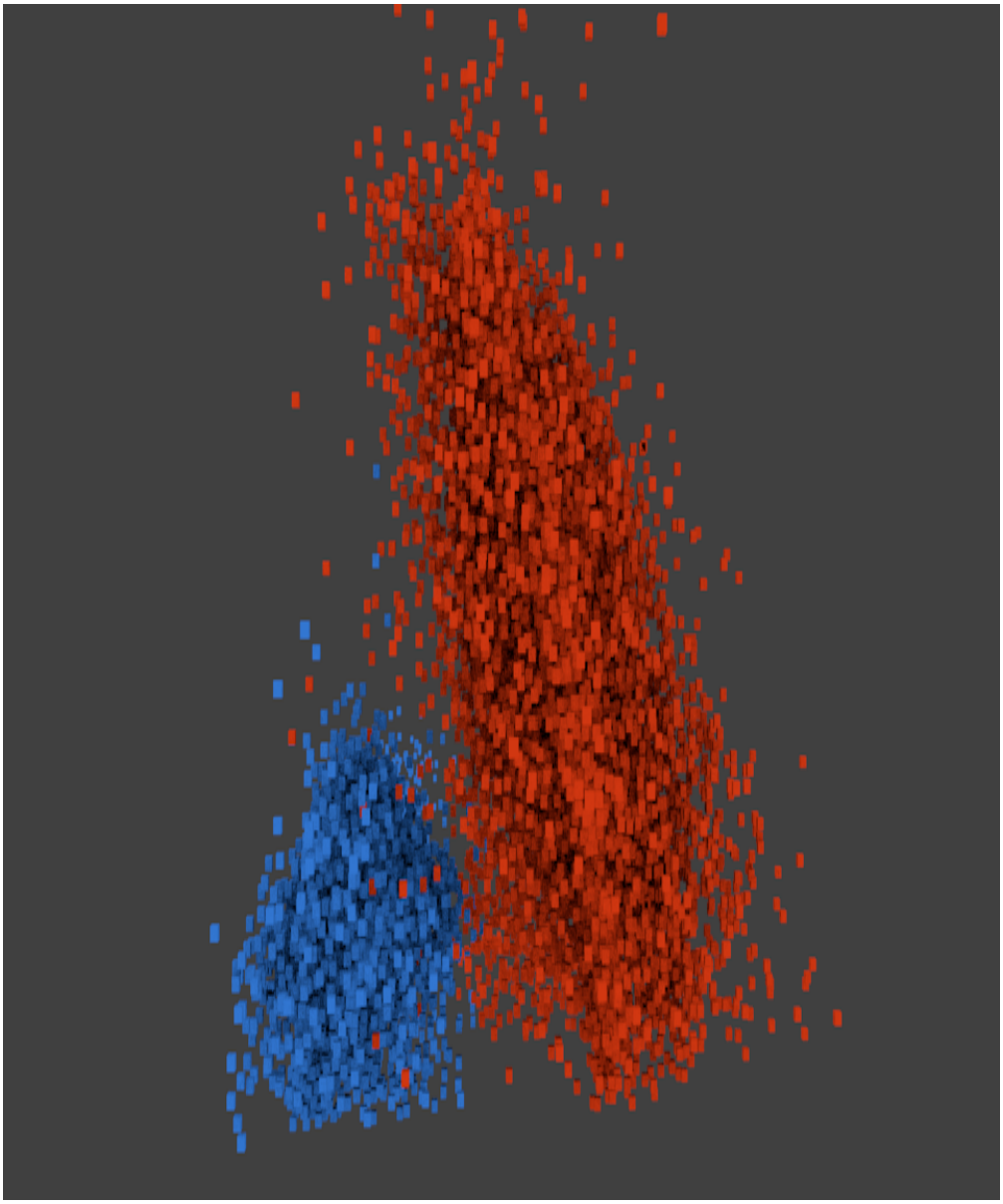
Dec 18, 2022

User Documentation

1 Getting Started **3**
- 1.1 Getting the data . . . 3
- 1.2 Creating the model . . . 3
- 1.3 Loading the model into Blender . . . 4

2 More Features **7**
- 2.1 Plotting Categories . . . 7
- 2.2 Setting Number of Rows to Plot . . . 8
- 2.3 Changing the Point Size . . . 8
- 2.4 Changing the Spacing of the Points . . . 8

Blendplot is a Python application for generating 3D model scatter plots of data to use in the 3D modeling software Blender. It can take `.csv` data files with labeled columns and create `.obj` models of scatter plots using the specified columns.



CHAPTER 1

Getting Started

To introduce you to Blendplot, we will show how to take a simple dataset, plot it using Blendplot, and then load the created model into Blender. The dataset we will be using is a dummy dataset for products sold in a supermarket.

1.1 Getting the data

This dataset contains information on items and includes their product name, cost, weight, height, quantity, and type. The dataset is defined below.

```
"product","cost","weight","height","quantity","type"
"milk",8.99,10.0,2.0,25,"food"
"chocolate",5.99,15.2,0.5,10,"food"
"bannana",2.99,2.0,0.5,20,"food"
"television",199.99,100.0,10.2,3,"electronic"
"laptop"149.99,5.0,1.0,5,"electronic"
```

Copy the above data and save it into a file named `items.csv`.

1.2 Creating the model

Now that you have the data file, you can begin plotting it. First go to the directory where you saved the `items.csv` file in the command line.

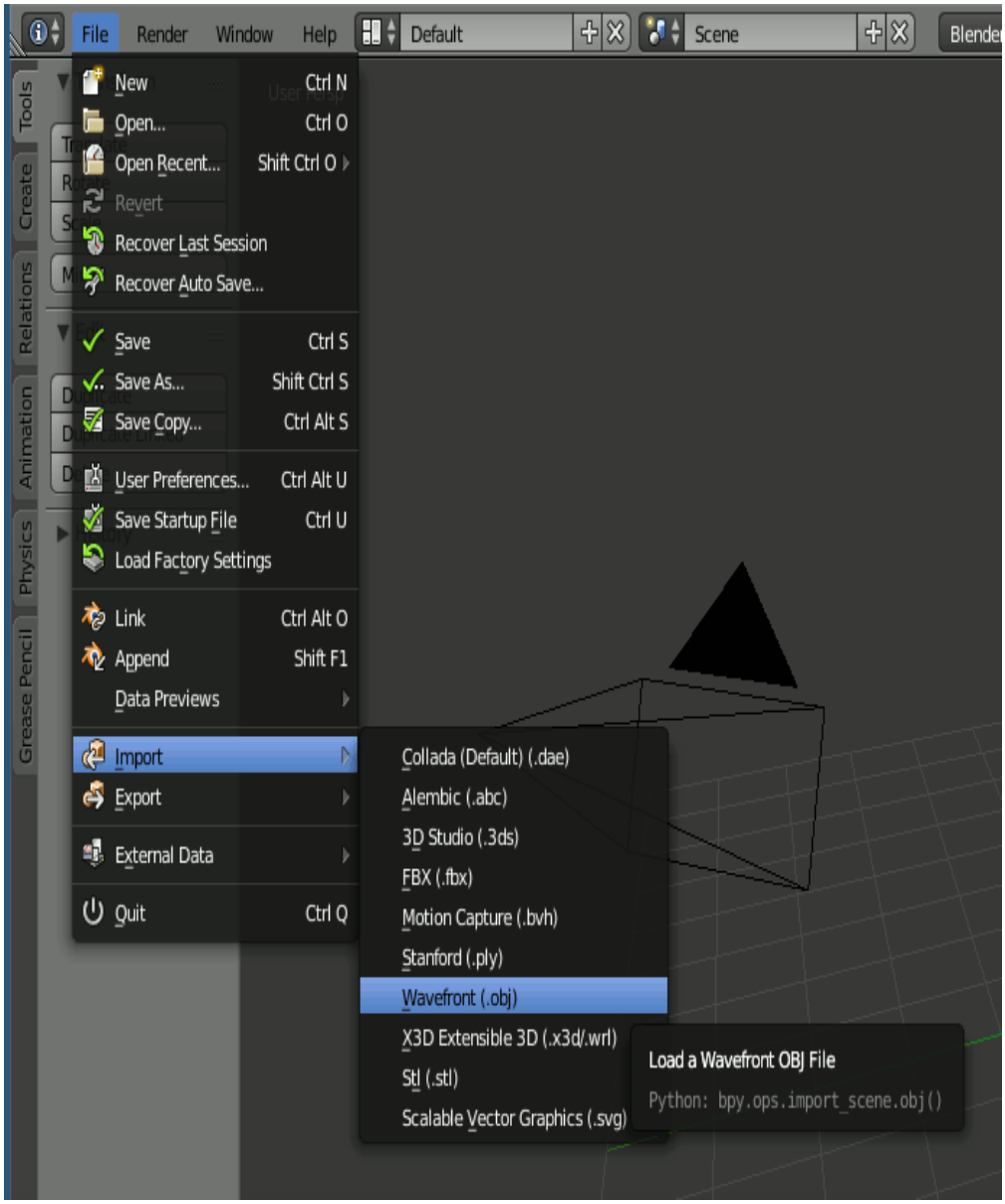
To begin, we will just plot the cost, weight, and height of the items. You can create the model for this plot in a file `plot.obj` by running the following command.

```
$ blendplot items.csv plot.obj cost weight height
Wrote plot file to plot.obj
Plotted 5 points in 0.017454 seconds
```

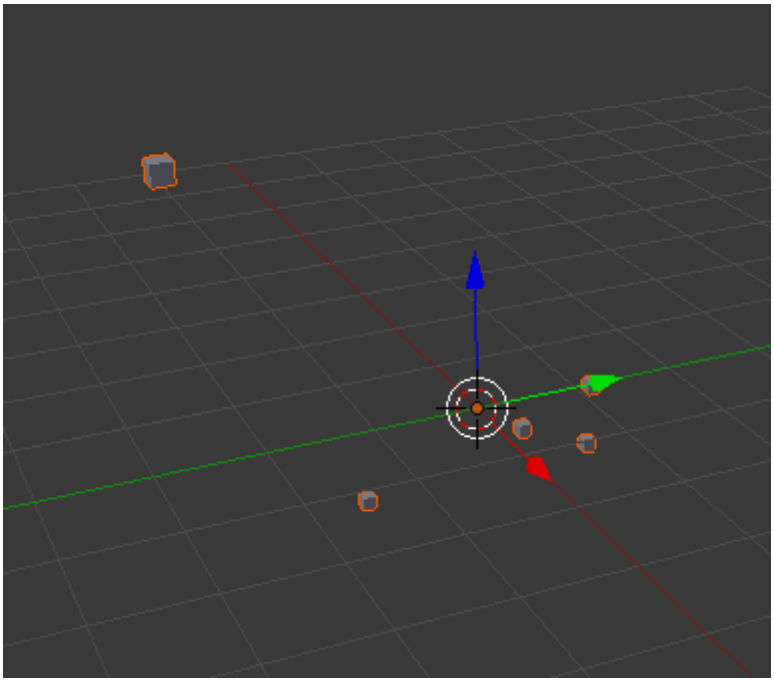
1.3 Loading the model into Blender

Now that you have created the model file `plot.obj` you can now import it into Blender.

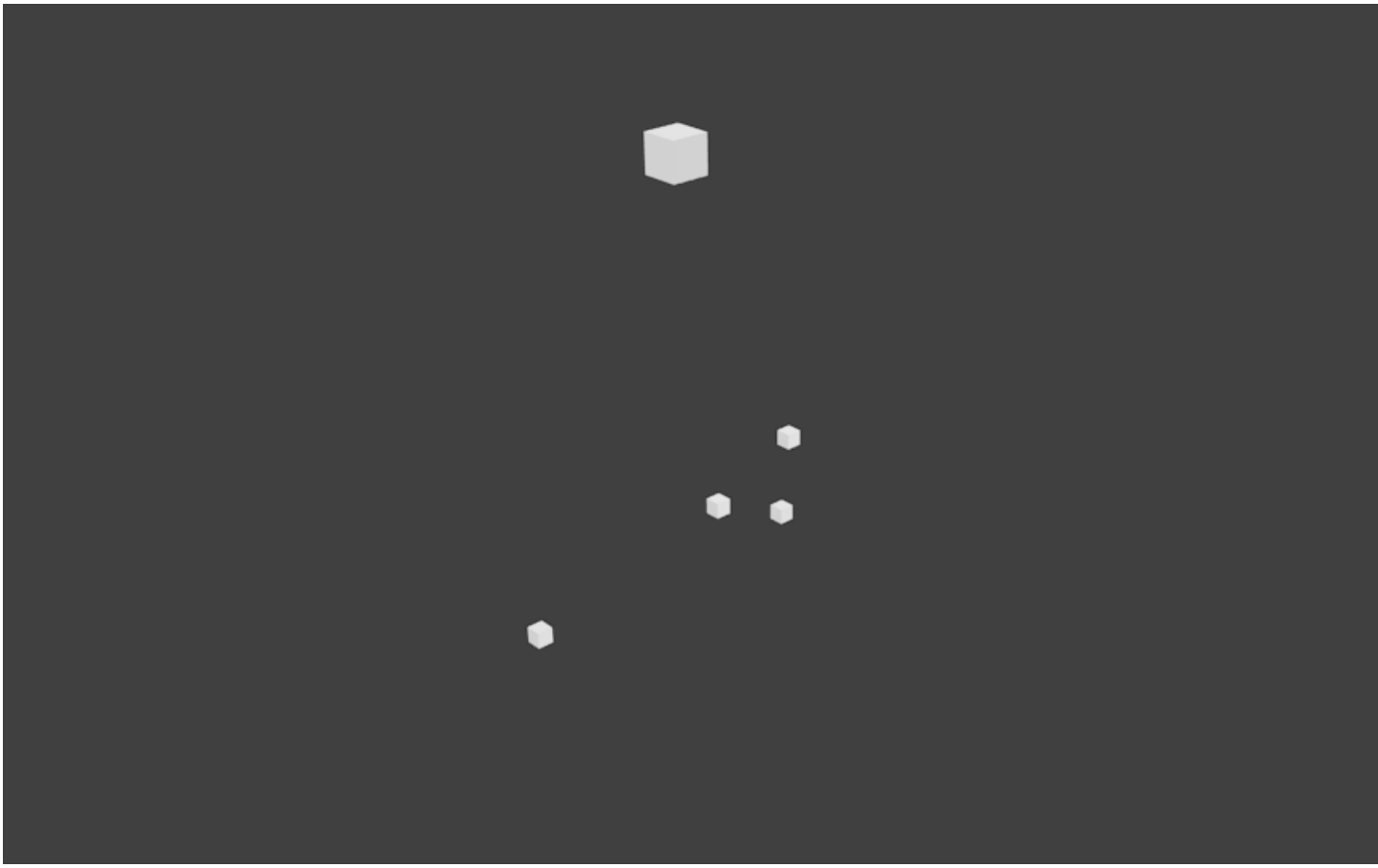
When you start up Blender, first delete the starting cube. Then go to `File > Import > Wavefront (.obj)` and select the `plot.obj` file that you created.



Once you have imported the model file, you should now have an object consisting of the plotted data points.



Now that you have the plot object you can make any changes you want to the object materials and scene and create rendered images or animations using the data.



This dataset was a bit small, but Blendplot supports very large datasets as well. Feel free to try it with your own

datasets and see how they look.

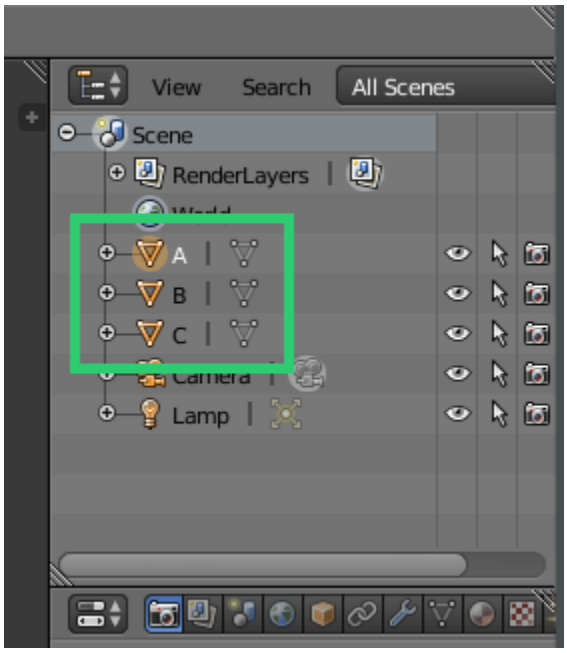
CHAPTER 2

More Features

2.1 Plotting Categories

To plot data that is grouped into categories, you can use the -c flag and specify the name of the column to group the data points by. Each different category will be its own object in the resulting model. This will allow you to easily style the groups with different materials.

```
blendplot data.csv model.obj height weight cost -c category
```



2.2 Setting Number of Rows to Plot

If you have a particularly large dataset, but only want to plot some of the points then you can use the `-r` or `--rows` flags to specify how many rows you want to plot. When you use the flag and specify a number of rows `n`, then only the first `n` points in the dataset will be plotted.

For example, the following command will plot only the first 20 points in the dataset.

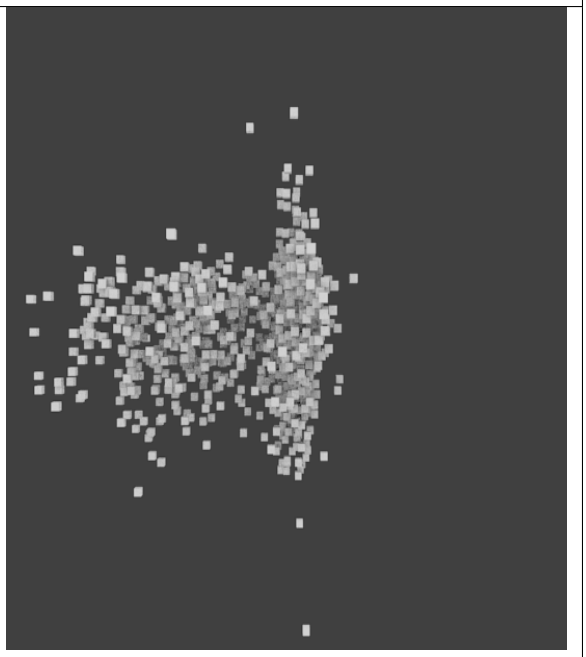
```
blendplot data.csv model.obj height weight cost --rows 20
```

2.3 Changing the Point Size

If you have data that is particularly dense or spread out and want to change the size of the cubes generated for each point, then you can use the `--pointsize` flag. When you use the flag and specify a point size, then all of the points will be of that size. The default point size is `0.0625`.

For example, the following command will increase the size of the points to `0.2`

```
blendplot data.csv model.obj height weight cost --point-size 0.2
```

0.0625	0.2
	

2.4 Changing the Spacing of the Points

If you want to change the spacing between the points, then there are two related flags that you can use to change it `--scale-function` and `--spacing`.

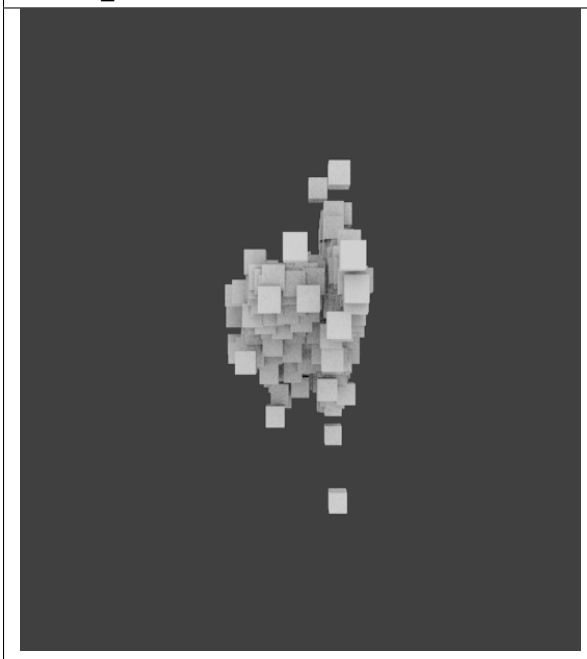
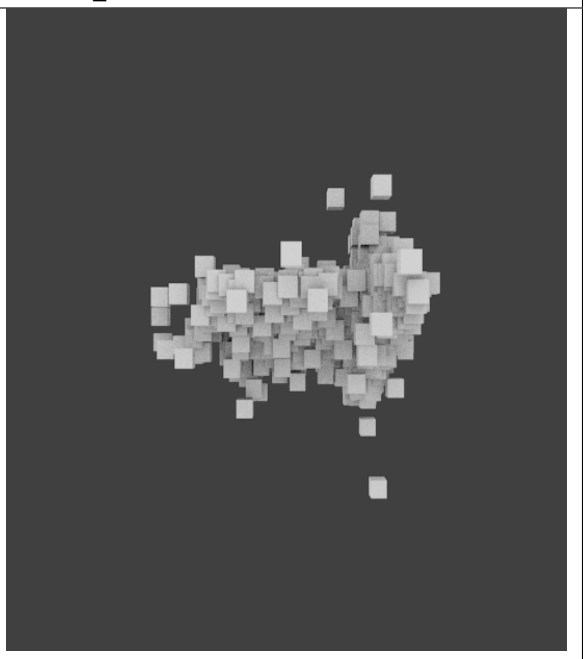
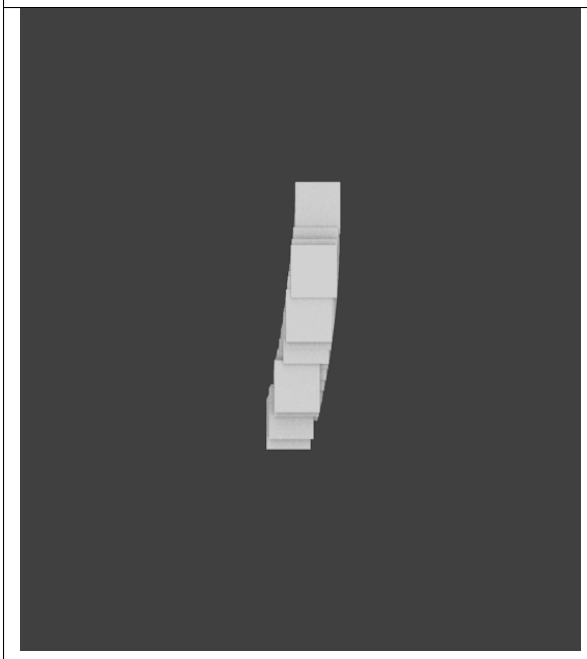
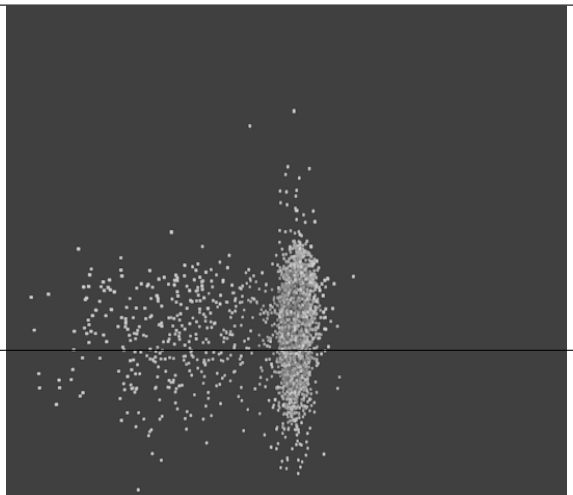
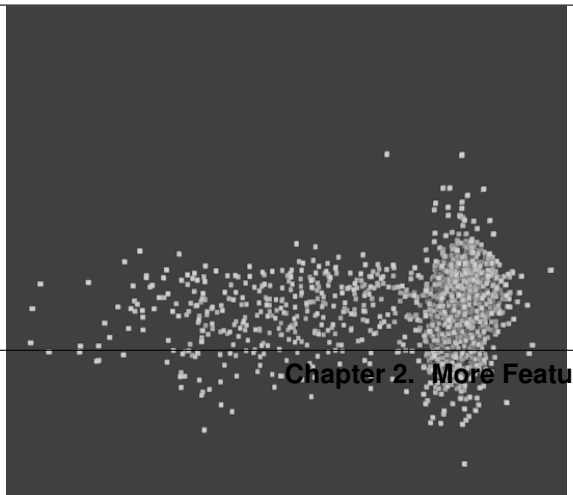
2.4.1 Scale Function

Blendplot applies a scaling function to the input data to get it tofit into a reasonable space and account for differences in the magnitude of different variables. By default, it uses the `preprocessing.scale` function from the `sklearn` library, however using the `--scale-function` flag it can be configured to use other scaling functions or no scaling function at all.

For example, the following command will plot the dataset using the `normalize` scaling function.

```
blendplot data.csv model.obj height weight cost --scale-function normalize
```

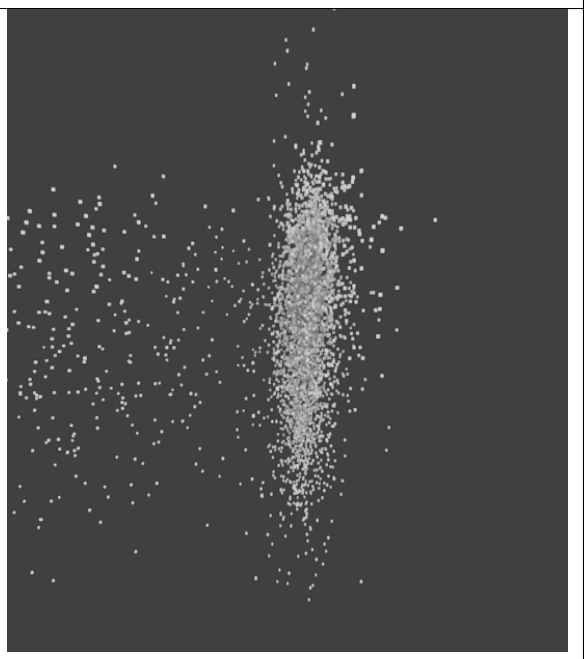
The following scaling functions are supported.

maxabs_scale	minmax_scale
	
normalize	robust_scale
	
scale	none
	

2.4.2 Spacing

You can also change the point spacing by setting a spacing ratio using the `--spacing` flag. The (x,y,z) coordinates for each point are multiplied by the spacing ratio in order to control the spacing between data points. By default, a spacing ratio of `2.0` is used.

For example, the following command will plot the data with a larger spacing ratio of `4.0`.

```
blendplot data.csv model.obj height weight cost --spacing 4.0
```

2.0	4.0