using visible light spectrum to cluster relative distributed, ordered collections of datapoints

by Dirk Biesinger, BA BIS, AIMT | saravjishut.org

June 2022, Rev 1

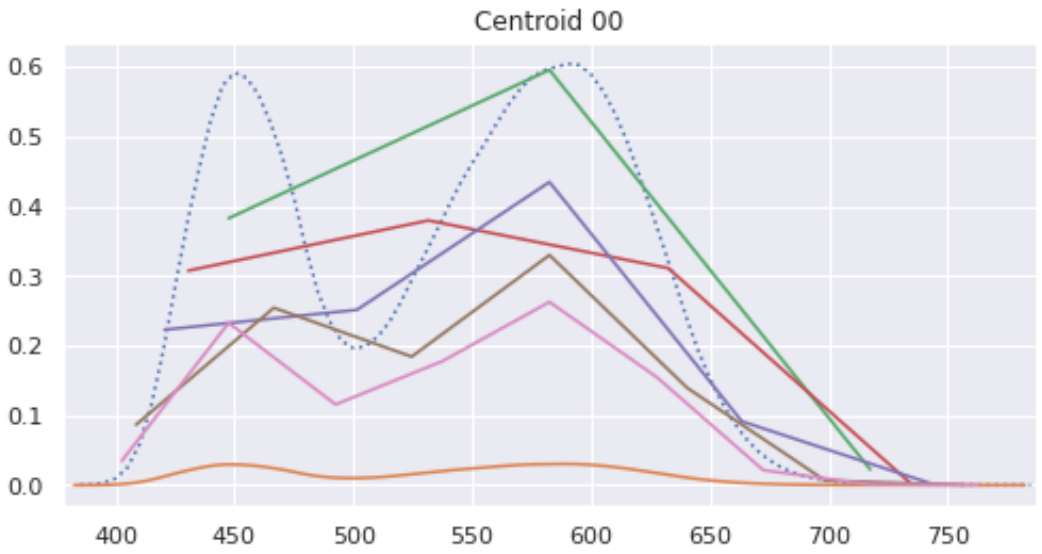
Executive Summary

Relative distributed, ordered data as found e.g. in sentiment analysis, packaging size, garment size, relative performance in technology products or quality ratings is difficult to analyze the moment the number of elements are not the same across all the data.

The relative distributed refers to the percentage of all elements combined equal 100%. The ordered refers to some form of measure in which the elements are arranged from minimum to maximum.

In this paper I show how transforming the relative distributions into color codes and use this well established and researched field to cluster similar distributions -independent of their number of elements.

I further show that the principal characteristics of these clusters can be extracted and converted into relative distributions of varying element numbers:



The blue, dotted line is a 20x version of the principal relative distribution in orange. The green line shows the relative distribution of 3 elements, red is 4 elements, purple is 5 elements, brown is 7 elements, pink is 9 elements.

Summary

Relative distributed, ordered data is present in many industries and analytical problems. To make these diverse observations comparable and usable for clustering, the idea of converting the data into visible light spectrum is explored.

I can show that it is not only possible to use characteristics of the optical light spectrum to compare and cluster the data, but that it is also possible to reverse the approach to use the found cluster centroids to generate representative distributions.

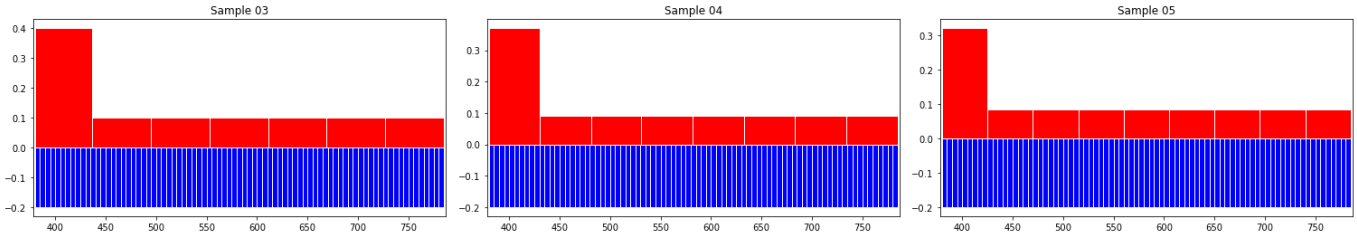
Introduction

In an ever more data driven and data conscious world, collecting and storing data is no longer a problem. The area of concern has shifted to connecting the data and using the data to glean insights out of it.

One of the general problems in connecting data is varying formats of the data on many levels.

A different challenge lies in how to present and codify what is a simple matter for humans but is in fact is relatively complex on a data basis.

Take the following illustrations as an example:



The characteristic of the representation looks very similar or comparable.

But: these are vastly different from a data point of view: All have a different number of items in the red data. There are 7, 8 and 9 in “Sample 03”, “Sample 04” and “Sample 05” respectively.

These differences pose a challenge when trying to use the data together.

Some Examples where this is applicable:

- data related to sentiment analysis where demographic information is available and the analysis spans multiple data sets based on different rating scales, e.g. 5 star rating and 7 or 10 step satisfaction rating
- packaging size in the food industry
- garment size in clothing
- relative performance in technology products (e.g iPhone 5 had 9 variants, iPhone 12 has 12 variants, iPhone 13 has 7. Or same example on a time basis: in 2013 there were 5, 5, 5, 6 variants available (per quarter), This changed to 20, 20, 20, 17 in 2021 (per quarter).
- quality rating e.g. in meat products

Is it possible to make this relative distributed, ordered data with differing number of elements comparable?

It occurred to me that there is another area that has similar characteristics: Converting a visible light spectrum curve to a color code. This is also a n:1 relationship: Many different light spectrum profiles can represent the same color code. Further, the light spectrum profile curve can be broken down into a composition of three distinct curves for red, green and blue.

Idea

Could the visible light spectrum and its methods for transformation be used to make relative distributed, ordered data comparable?

I set goals for evaluating this idea:

1. Explore if it is possible to convert relative distributed ordered data to visible light spectrum and if there is value in doing so.
2. Explore if this information could be used to create clusters. This included creating centroids and assigning a distinct color code based on this.
3. Explore if there is a way to reverse from the color information assigned to a centroid to a visible light spectrum curve and use this to create relative distributions for bucket qty n, where n is determined by the requirements from the incoming data.

This evaluation limits its scope to exploring the possibility to use visible light spectrum to compare relative distribution over different number of buckets. As the visible light spectrum to color code is a n:1 relationship, this evaluation also attempts to find one solution to generate a visible light spectrum out of a color code.

This evaluation will not assess how much of an influence these results have in predictive models.

Method

One solution to this problem is to convert the individual subsets of the data belonging to one relative distribution into visible light spectrum, from there into a color code and x-y location on the CIE standard chromaticity diagram.

To achieve this, 5nm bucket size in the visible light spectrum from 340-780nm | 81 buckets is used as an approximation.

This then allows us to compare the relative distribution characteristics of each collection without regard to the number of sub-elements.

This in turn lends itself to cluster distributions of similar characteristics. This can be used to evaluate or model the data based on these clusters.

Additionally, collecting the centroids of clustering algorithms yields an x-y coordinate in the CIE standard chromaticity diagram. This coordinate can be converted back to a distribution curve based on the above used approximation of 5nm intervals.

This relative granular distribution then can be converted into n-buckets as required by the original data by calculating the area under the distribution curve for each final bucket. The last step then is to normalize to 100% to again achieve a relative distribution.

Results

Observation

Assumptions about the data: The data is prepared in a manner that includes a relative distribution array (sum = 1.00) of the individual buckets in the right order (from smallest to largest or lowest to highest). At this stage, the length of these arrays equates to the number of

buckets. Zero values specifically are required to be included. (No implicit information due to lack of data is feasible)

Using a “Toy” min Sample Dataset:

A minimum example of the required data:

li_01 = [0.2, 0.3, 0.5]

li_02 = [0.15, 0.2, 0.25, 0.25, 0.15]

li_03 = [0.4, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1]

li_04 = [0.1, 0.1, 0.1, 0.4, 0.1, 0.1, 0.1]

li_05 = [0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.4]

li_06 = [0.7, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05]

li_07 = [0.05, 0.05, 0.05, 0.7, 0.05, 0.05, 0.05]

li_08 = [0.05, 0.05, 0.05, 0.05, 0.05, 0.7, 0.05]

li_09 = [0.22, 0.32, 0.46]

The next step is to determine the edges of the bucket space for each observation. Visible light spectrum is typically defined in the range [380, 780]nm. The calibration tables available and included in the project code list values for 380 to 780nm. This method uses buckets of 5nm width as an approximation and simplification for calculations. With a 5nm bucket having a starting value and an ending value, a further approximation was done by assigning the available calibration data to a 5nm bucket with the starting value of said calibration data. This results in 81 5nm buckets with an effective coverage of the spectrum [380, 785]nm.

Consequently, the relative distribution of each observation needs to be equally spaced over the interval [380, 785]. At this step another approximation is done: the resulting edges are rounded to integers.

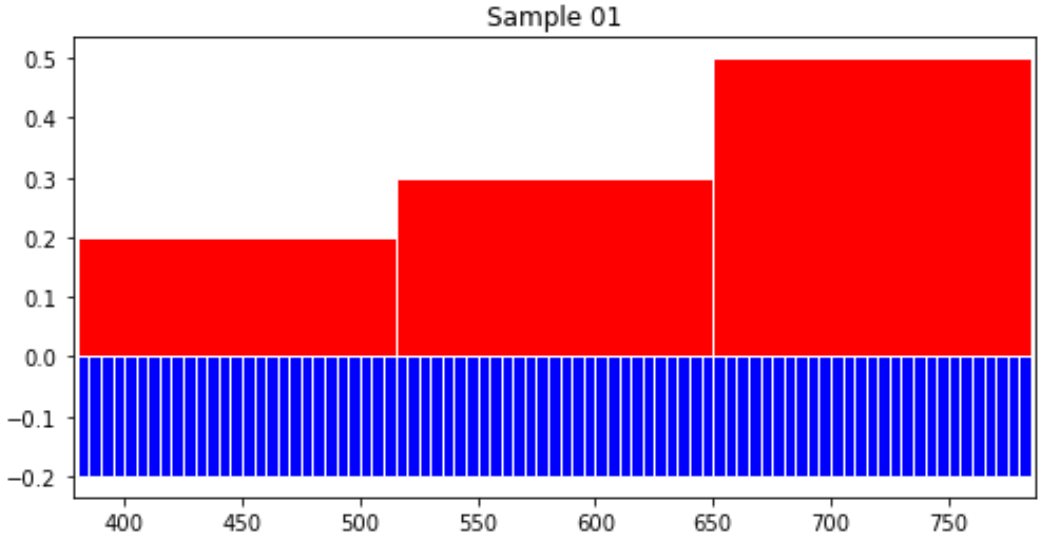
The resulting edges for two of the above observations:

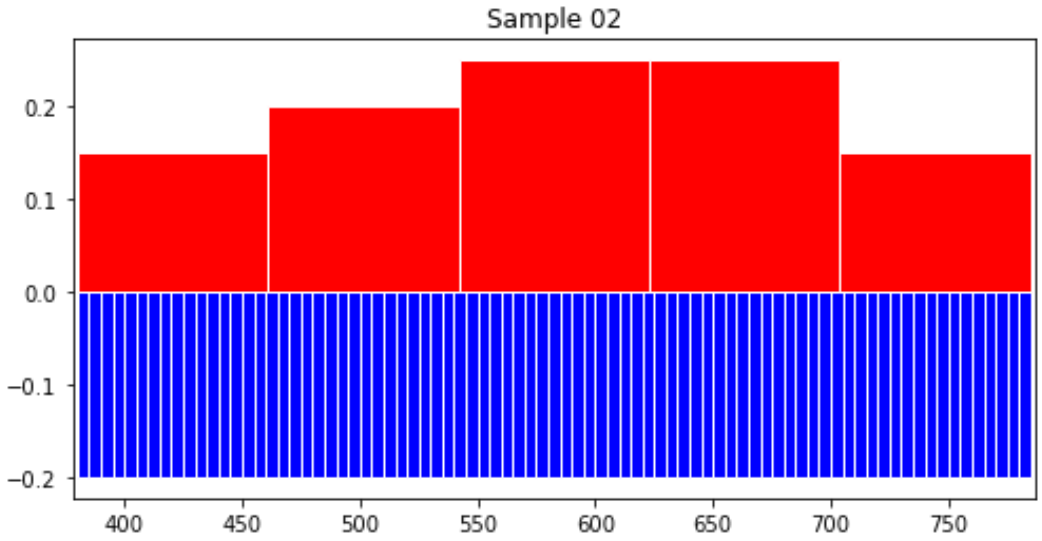
sample 01: [380 515 650 785]

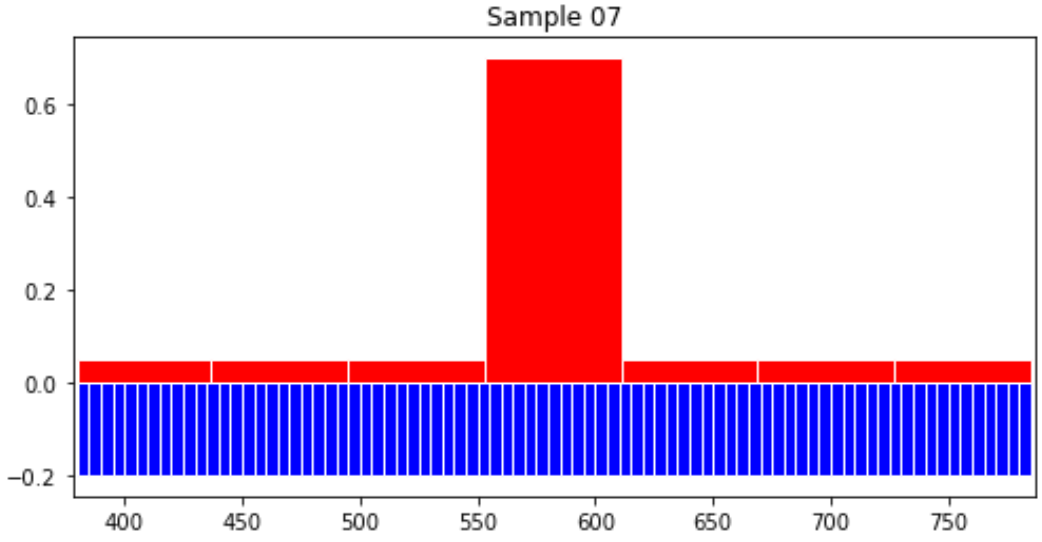
sample 07: [380 437 495 553 611 669 727 785]

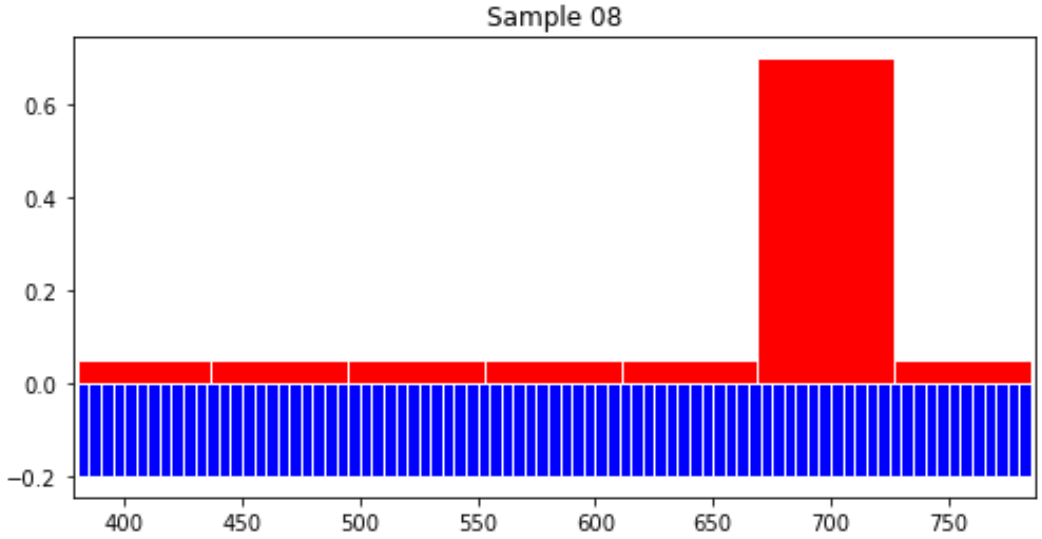
for reference, the edges in the evaluation space in 5nm buckets : [380, 385, 390, 395, 400, 405, 410, 415, 420, 425, 430, 435, ..., 750, 755, 760, 765, 770, 775, 780, 785]

Combining the observation buckets with relative values with the 5nm evaluation range buckets yields these visuals (select examples, corresponding to above sample data numbering):

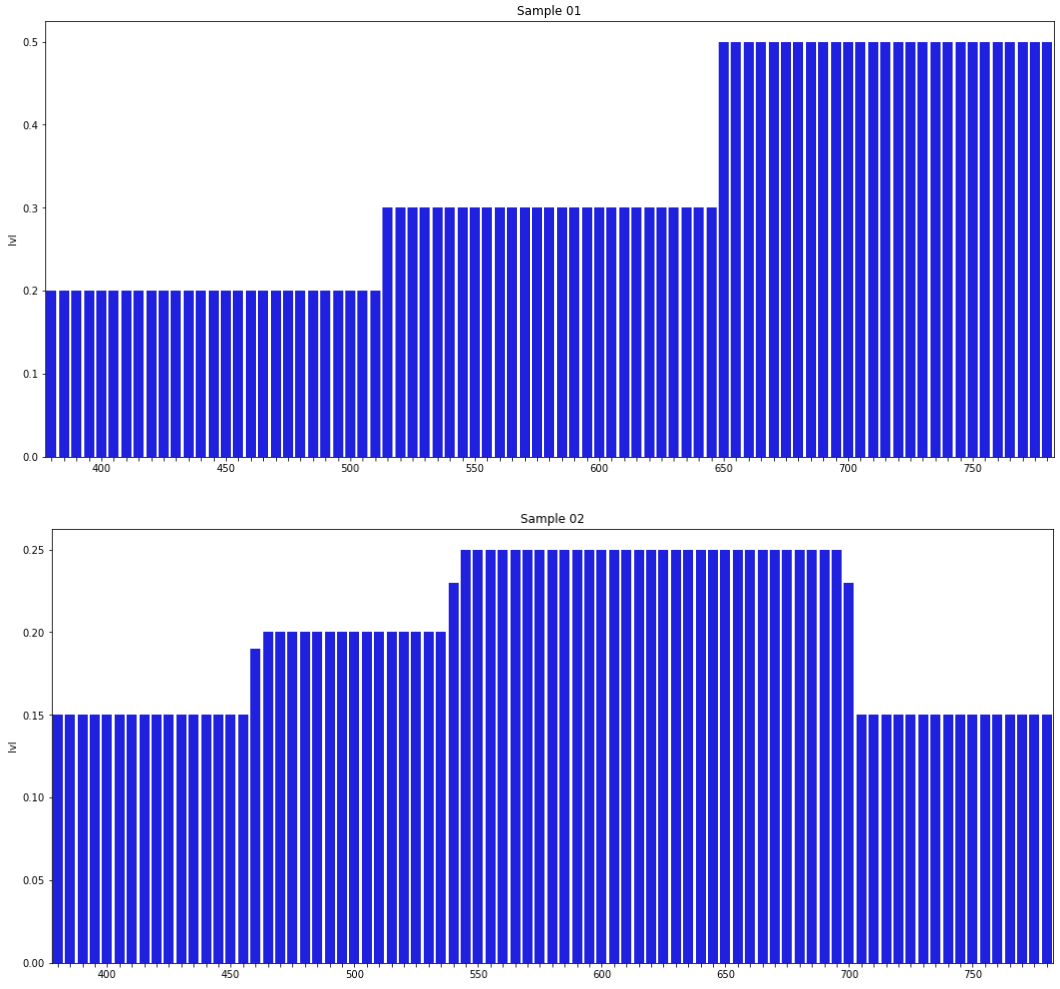

Sample 01
0.5
0.4
0.3
0.2
0.1
0.0
−0.1
−0.2
400
450
500
550
600
650
700
750

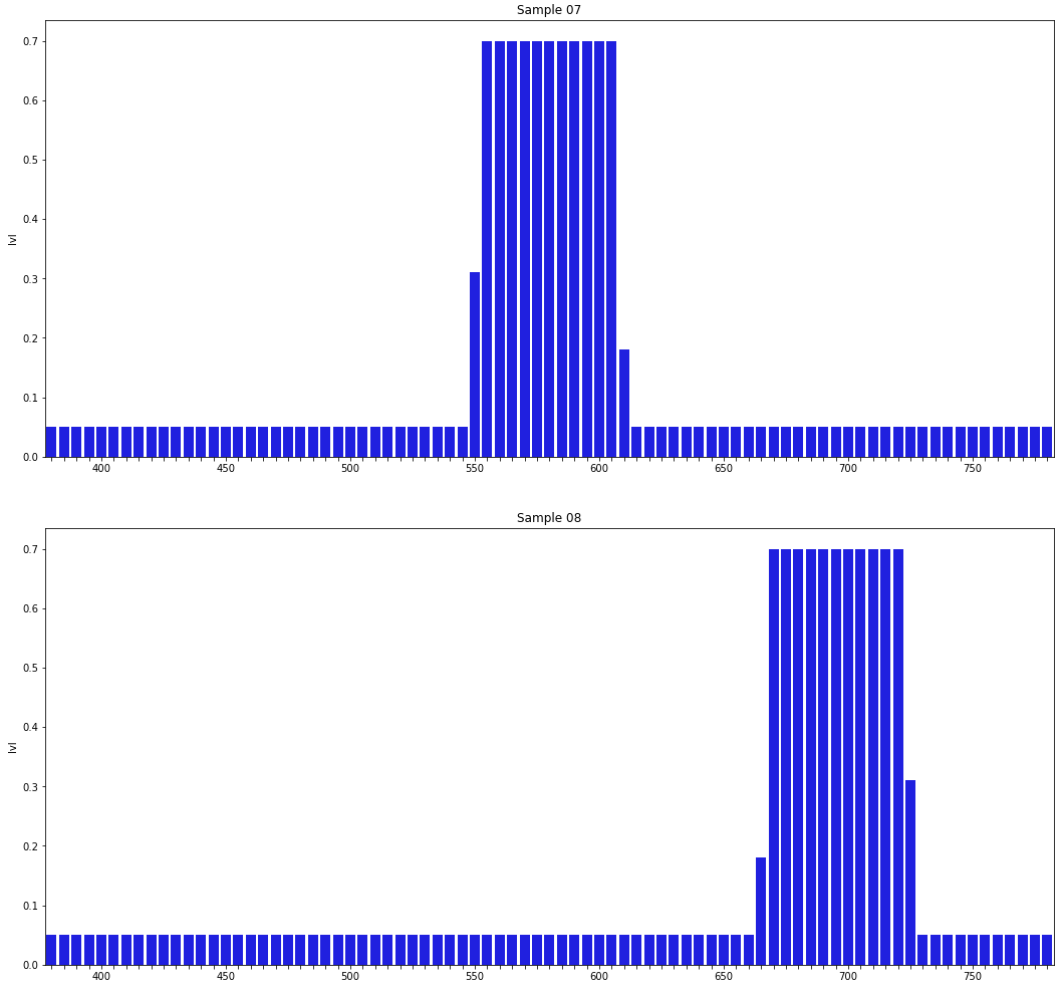

Sample 02
0.2
0.1
0.0
−0.1
−0.2
400
450
500
550
600
650
700
750


Sample 07
0.6
0.4
0.2
0.0
−0.2
400
450
500
550
600
650
700
750

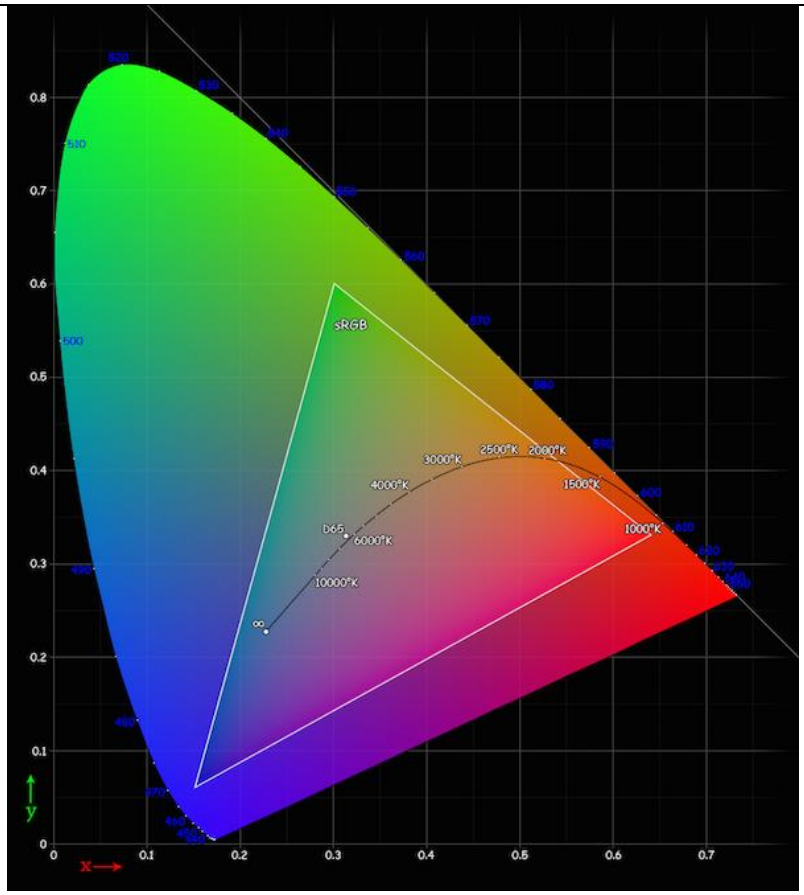


The next step is to calculate the value in each of the 5nm buckets. This is straight forward. The same as in the observation for all evaluation space buckets that do not contain an observation bucket edge. For the evaluation buckets containing an observation bucket edge, the calculation is to weight the two observation bucket values by the coverage in the evaluation bucket. This step yields a distribution in the 5nm bucket dimension that resembles the original observation distribution (note the interim steps at the observation bucket edges):

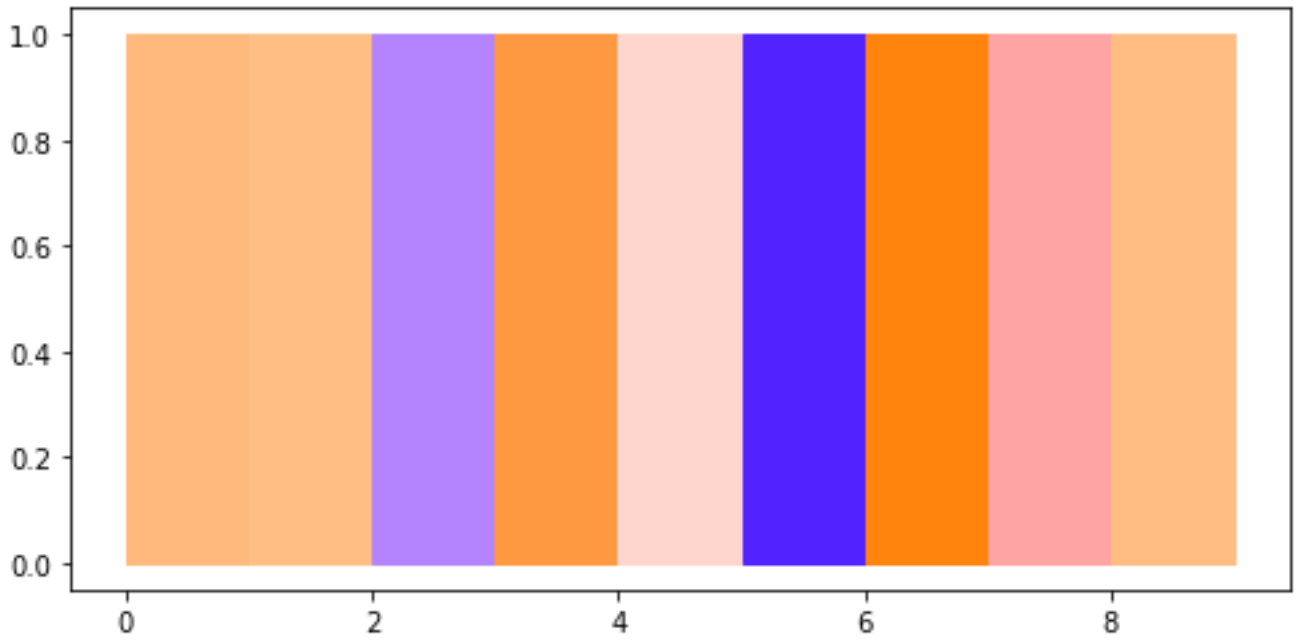




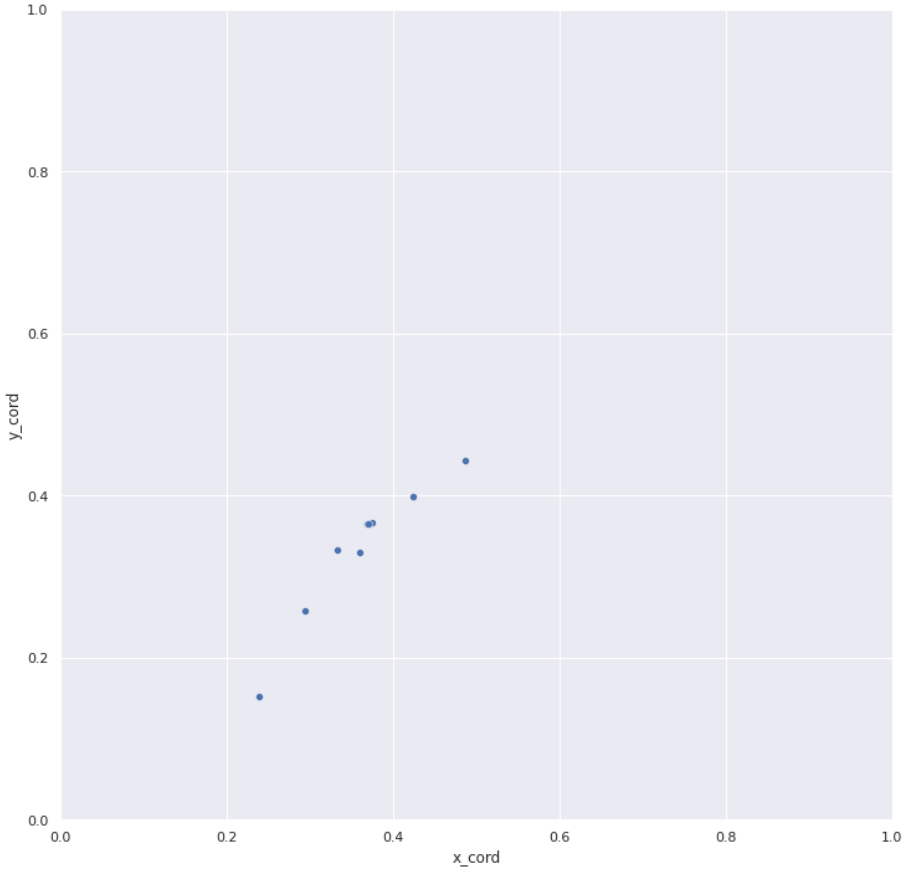
This information now can be used to generate color codes from this stylized visible light spectrum curve. These color codes can easily be transformed between RGB, HEX or x-y coordinates for the CIE standard chromaticity diagram:



Displaying the actual color by their respective codes, yields this from the above sample of 9 observations:



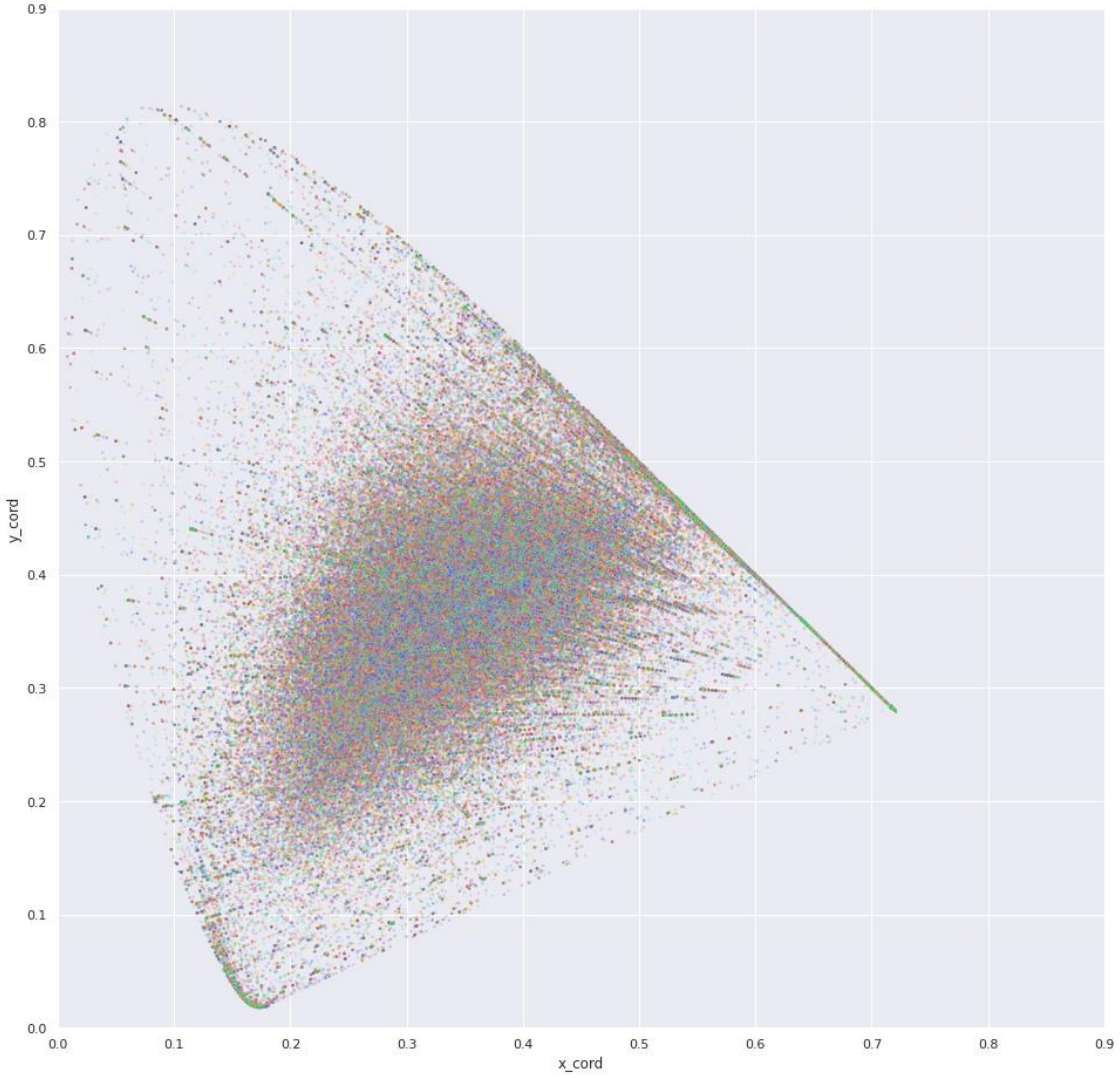
on the CIE plot, this data looks like:



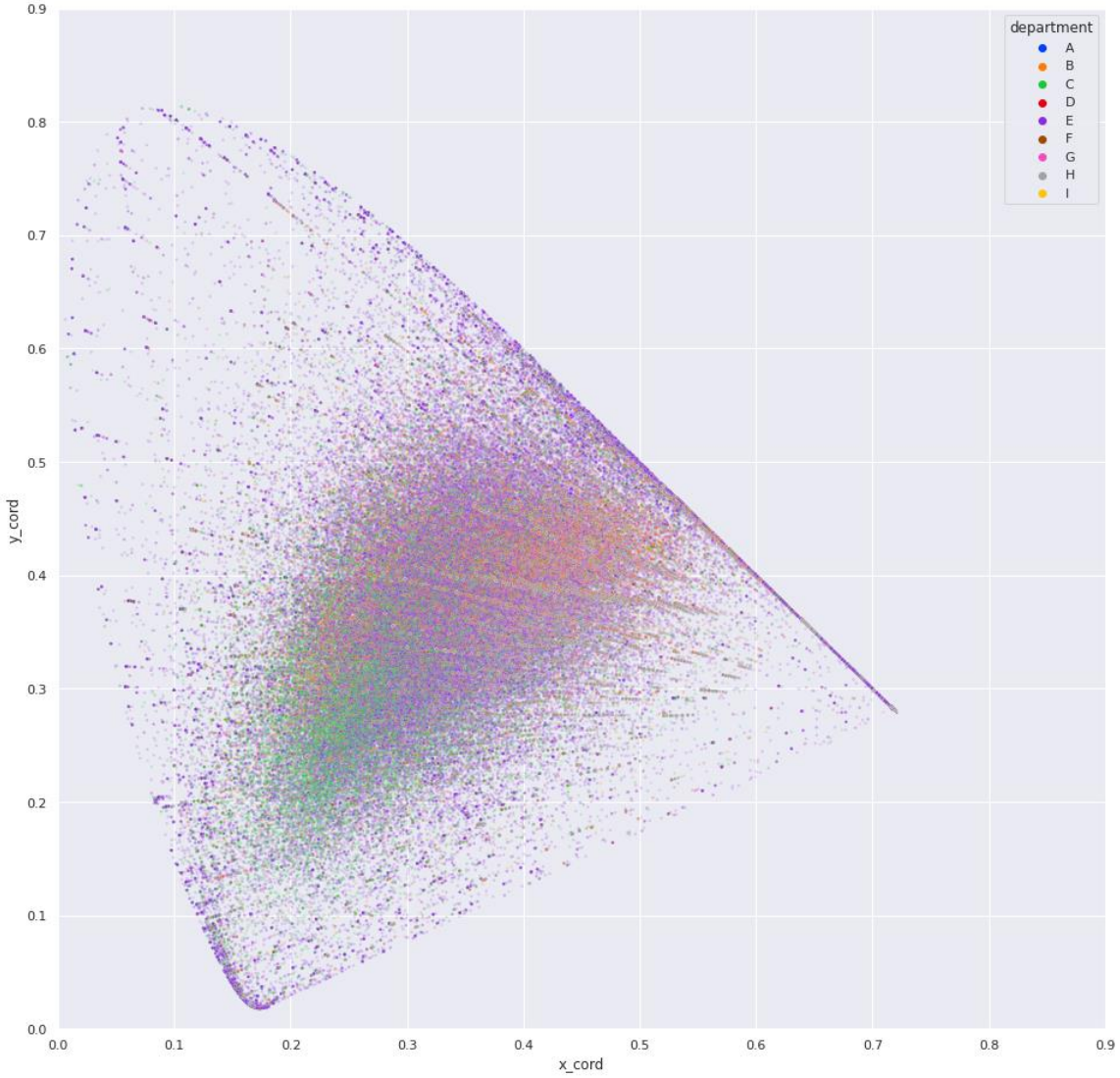
From this, it is immediately visible that there are distinct and different distributions as well as similar distributions in this sample set.

More Data

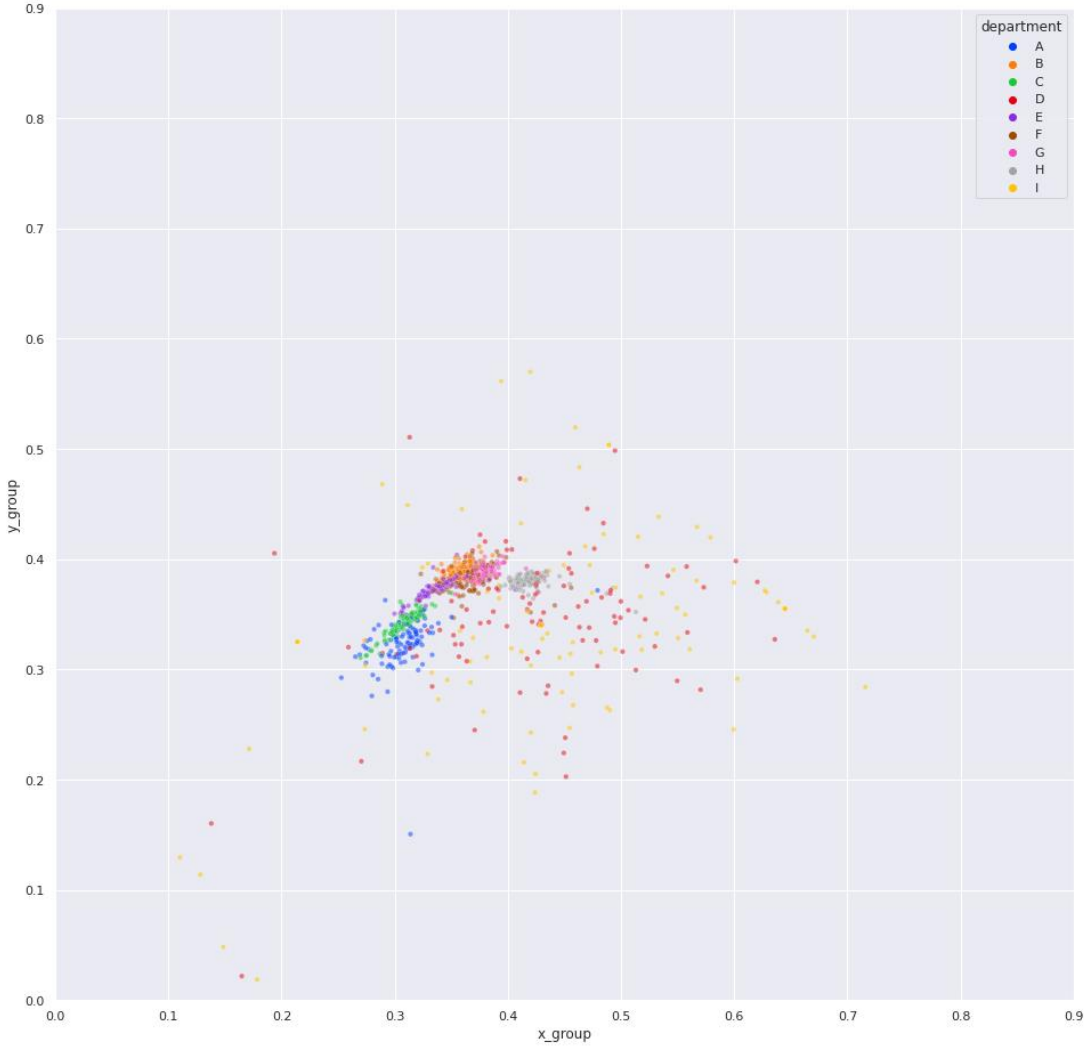
From this point forward a larger and richer dataset is used. This dataset, based on actual sales data, it is cleaned, anonymized, and preprocessed to combine the individual items into an ordered relative distributed array. The incoming data size was 2.1M observations in numerous relative distribution bucket configurations. Additionally, the data has 103 distinct locations, 9 distinct departments (with varying coverage between locations), 11,519 distinct articles in a total of 512,126 observations. This dataset is not only a real-world example, but also allows, due to its composition, for multiple dimensions of grouping. This dataset also contains the actual (absolute) sales quantities of the items. This can be used to create a weighted evaluation. In this dataset, the mean sales quantity is 12.32 items, while the max is 1,184 and median 7.



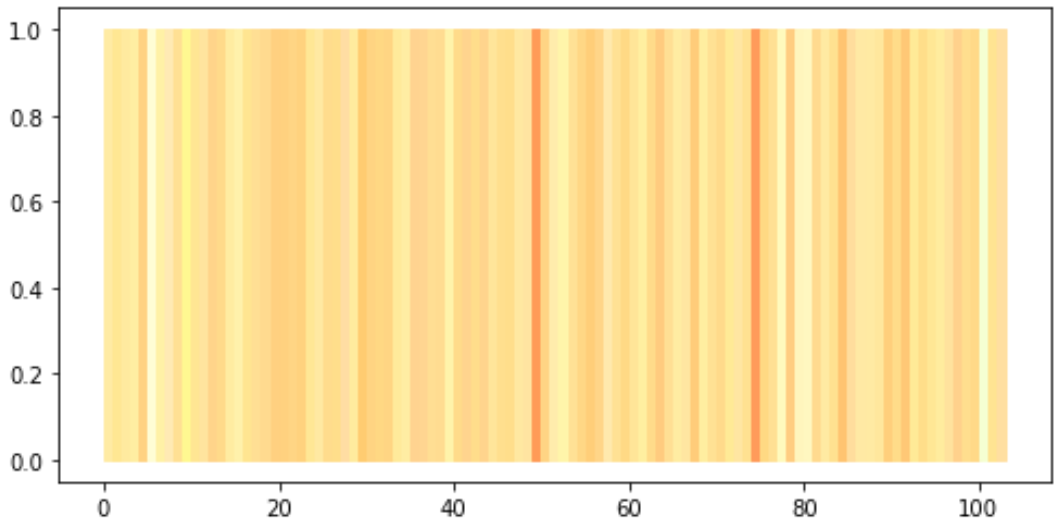
In the visualizations, lines appear. This is explainable by the numbers of buckets in the observations.

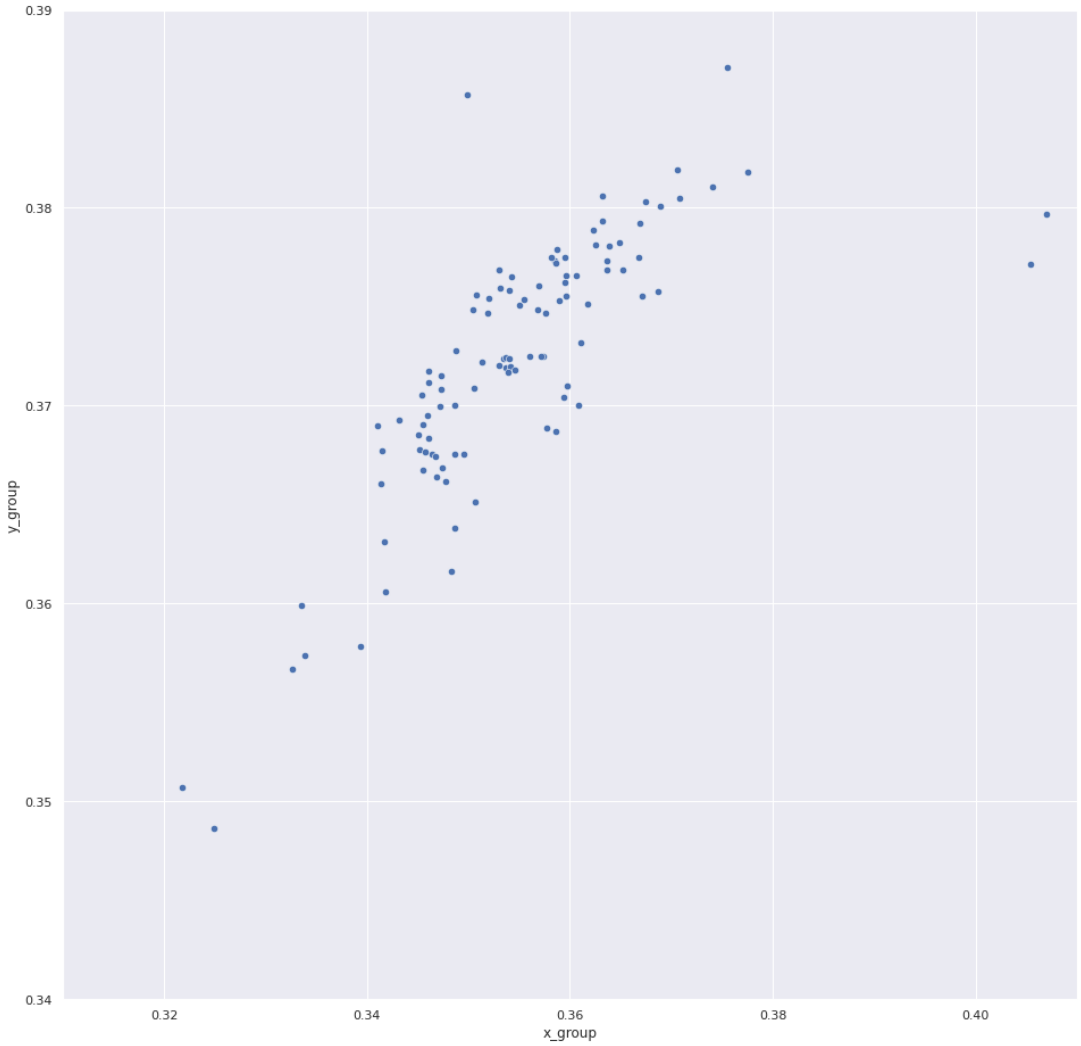


In appropriate groupings and color coding the plots based on these groupings, clusters appear.



The question, in which context this method was developed, concerned itself with the potential of clustering based on location:



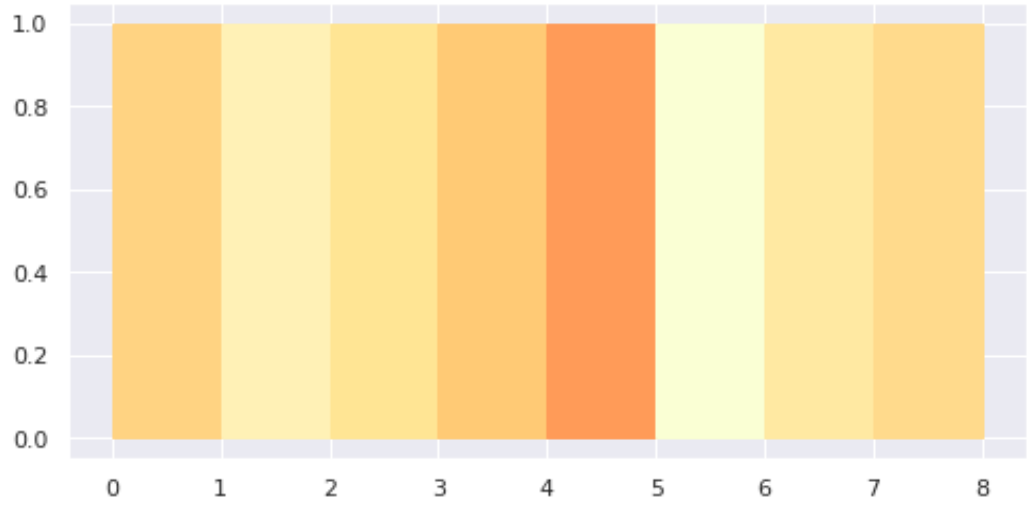


Using a clustering algorithm on the x-y coordinates of the resulting color-code when grouping by location yields multiple distinct clusters.

Using Gaussian Mixture with 7 clusters:

Using KMeans with 8 clusters:

Using the cluster centroids, a color code can be derived from the x-y coordinates:



Now, to extract a color spectrum curve is not a simple conversion, as there is a n:1 relationship between color curves to color code.

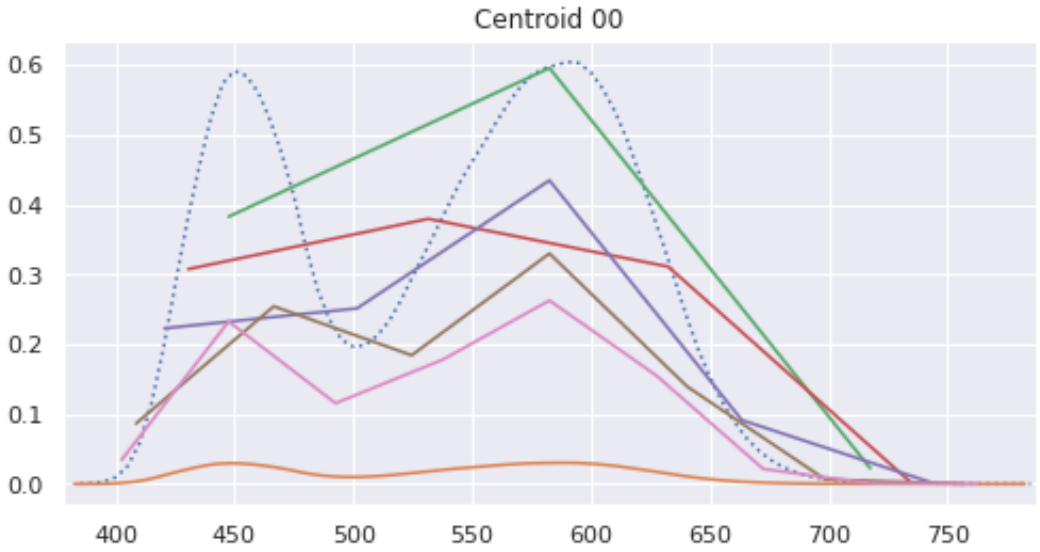
To get a color spectrum curve, some baseline needed to be used to anchor the n:1 relationship to a 1:1 relationship.

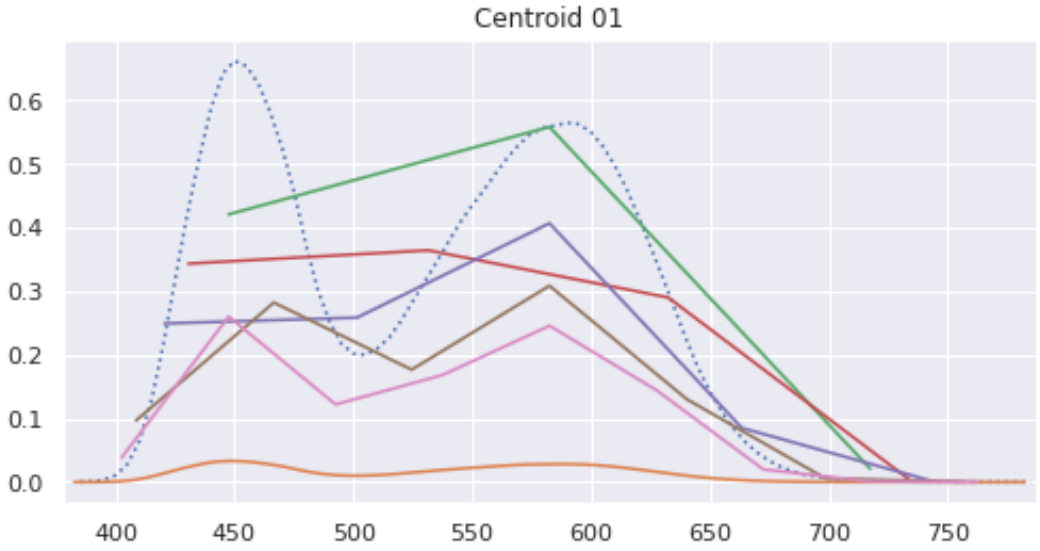
In this case, the Illuminnat D65 calibration is utilized.

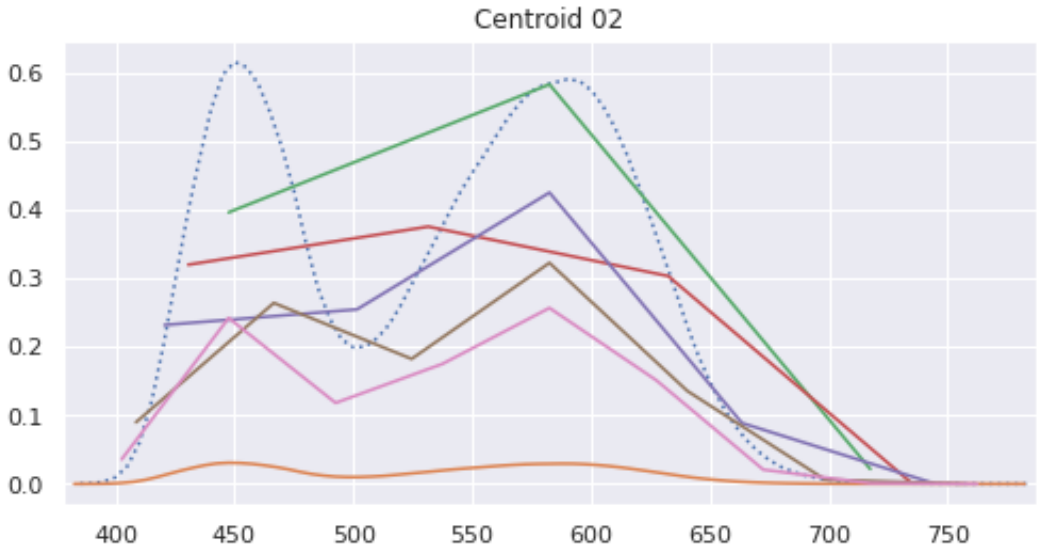
Based on this, a "curve" based on the 81 points on the 5nm edges is created.

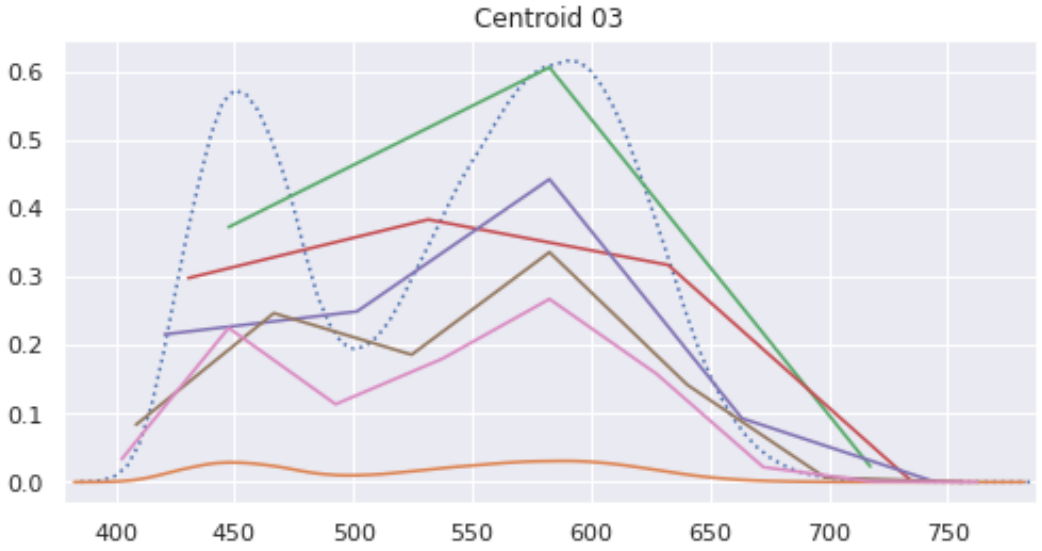
The last step is to reverse the attribution of the 5nm buckets into the buckets of the observation distributions. The principal idea is the same: sum up all the 5nm buckets that do not touch an observation bucket edge and add to that bucket. Treat the 5nm buckets touching an observation bucket with weighting based on the portions and add result to the appropriate observation buckets. The final step is to normalize the observation distributions to sum to 1.00.

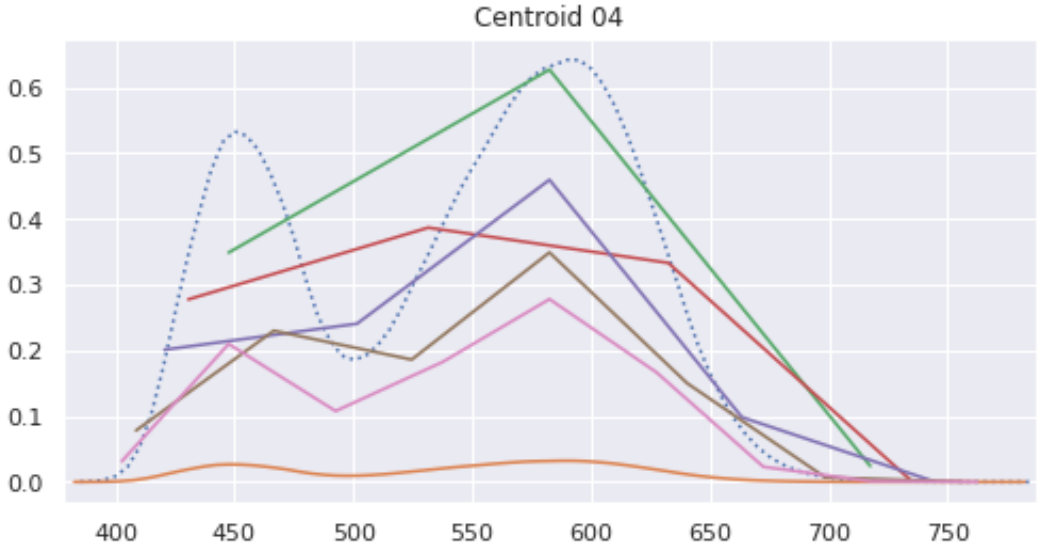
Please note: in the following graphs, the blue dotted line, is a magnified (by factor 20) curve of the 81 bin representation. The orange line at the bottom is the actual scale. All other lines are the results when applying 3, 4, 5, 7, 9 bucket distributions.

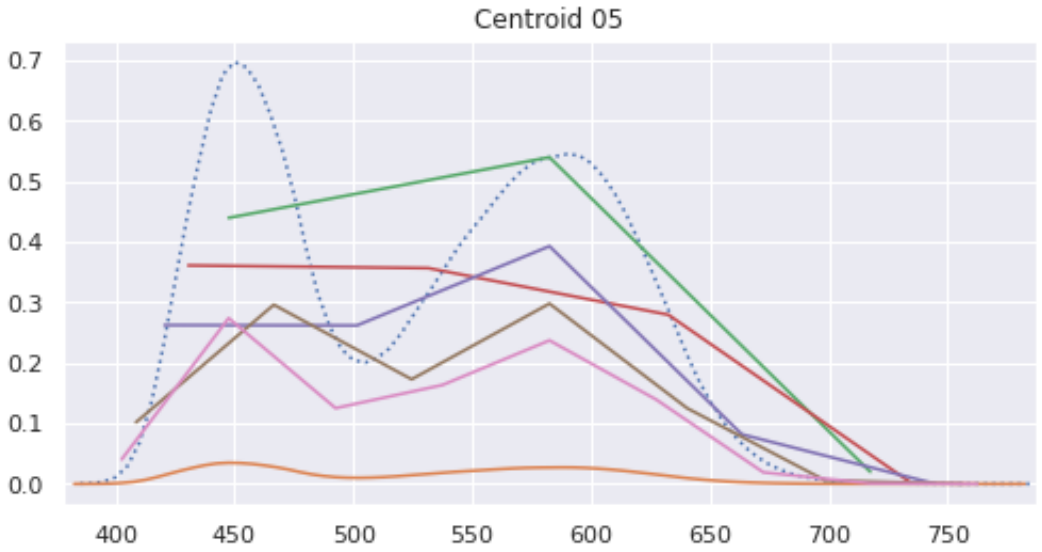


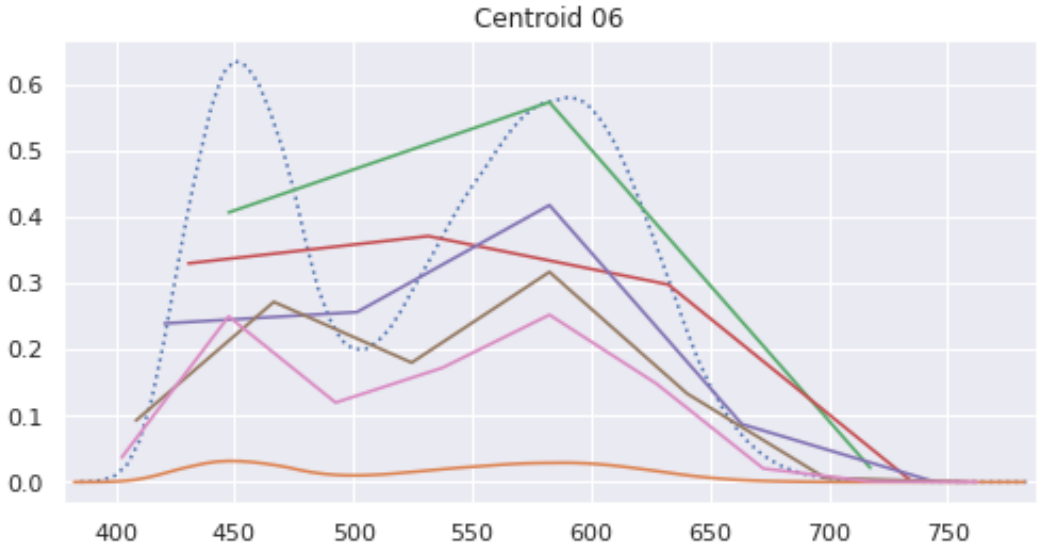


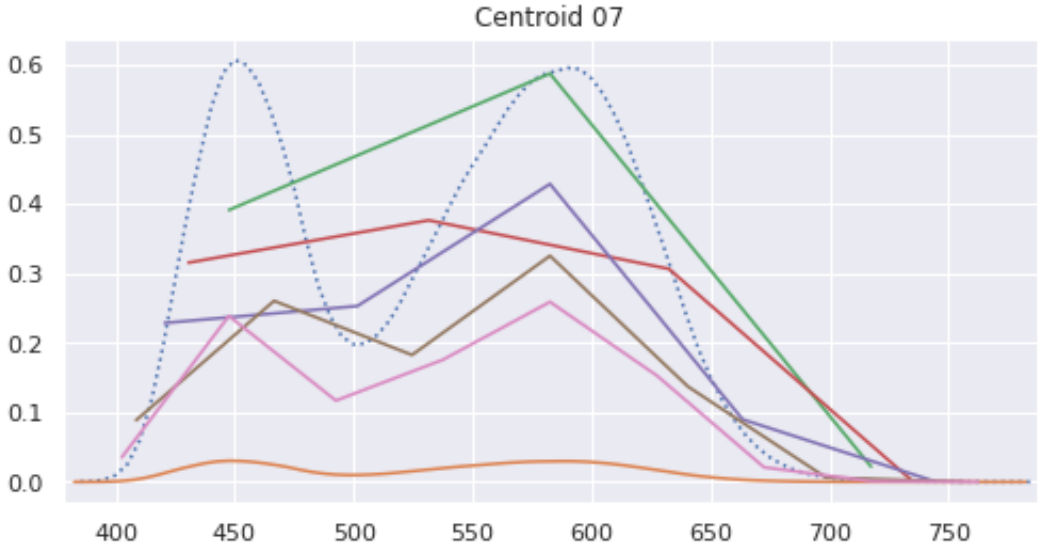

Centroid 02
0.6
0.5
0.4
0.3
0.2
0.1
0.0
400
450
500
550
600
650
700
750


Centroid 03
0.6
0.5
0.4
0.3
0.2
0.1
0.0
400
450
500
550
600
650
700
750


Centroid 04
0.6
0.5
0.4
0.3
0.2
0.1
0.0
400
450
500
550
600
650
700
750


Centroid 05
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
400
450
500
550
600
650
700
750


Centroid 06
0.6
0.5
0.4
0.3
0.2
0.1
0.0
400
450
500
550
600
650
700
750


Centroid 07
0.6
0.5
0.4
0.3
0.2
0.1
0.0
400
450
500
550
600
650
700
750

Discussion:

recap of goals:

The first goal was to explore if it is possible to convert relative distributed ordered data to visible light spectrum and if there is value in doing so.

The second goal was to explore if this information could be used to create clusters. This included creating centroids and assigning a distinct color code based on this.

The third and stretch goal was to explore if there is a way to reverse from the color information assigned to a centroid to a visible light spectrum curve and use this to create relative distributions for bucket qty n, where n is determined by the requirements from the incoming data.

The first goal was relatively easy to achieve: preparing the data in a suitable form and using the work done by Christian Hill.

It was possible to demonstrate a method of converting relative distributed ordered data to a color code and convert the color information between different color systems. Similar distributed data with a different number of data points per item achieves a similar result. Different distributions generate different results.

The second goal was achievable by expanding on C. Hill's work and appropriate data preparation work.

By expanding the achieved with appropriate data preparation work, we could demonstrate that we can explore data based on different groupings from within the data. Different patterns and distinct clusters emerge when exploring the data. The resulting data lends itself to clustering approaches and we could demonstrate replicable results and consistent results when using different clustering algorithms. We could generate individual clusters and assign data points to these clusters. With that, we achieved our original goal of being able to create clusters based on similar distribution characteristics.

To get to the third goal, the resolving the n:1 relationship from spectrum to color code is necessary.

The color codes are proportional to the components of a fundamental or calibration spectrum. The code package includes the CIE 1931 and CIE 1964 color matching functions and defaults to the 1964 version. This calibration is used for the conversion from spectrum to color codes, so we will be using the same basis for the conversion from color code to spectrum. We refer to this as matrix A.

Further, we need to choose an illuminant. The illuminant used for setting up and calibrating the conversion from spectrum to color code is Illuminant D65. We will use this same illuminant for the conversion from color ode to spectrum. We refer to this as matrix D65

Next, we need a scaling factor. We get this by applying the middle row of the transpose A' to D65:

s1 = 2315.81609165344

Next, we multiply our color code (in XYZ format) by the scale factor.

Then, we multiply this with our matrix A and sum to a 81 x 1 format.

This yields a distribution curve over the 81 buckets / 5nm scale.

These 81 buckets now can be assigned to the fewer buckets as required in a reverse manner of the initial assigning from the fewer buckets to the 81 buckets.

This allowed us to reverse the process starting from the cluster centroid to end up at a typical distribution curve that we then can convert into our relative distribution based on the number of required buckets. With this we can use this approach to inform a typical (average) distribution for a cluster.

References:

All illustrations but CIE standard chromaticity diagram by the author.

CIE standard chromaticity diagram: https://en.wikipedia.org/wiki/File:Cie_Chart_with_sRGB_gamut_by_spigget.png Original image by user Spigget, licensed under Creative Commons Attribution-Share Alike 3.0 Unported.

Christian Hill: Converting a spectrum to a color, 2016: https://scipython.com/blog/converting-a-spectrum-to-a-colour/

Demo notebook, source code, ref materials and sample data set: github.com/Saravji/cluster_by_color

Electronic version: saravjishut.org/