Exercise

Drawing inference and insight from diabetes data

Julian Flowers

2024-04-23

Table of Contents

## Introduction

In this exercise we will be using data from the English Public Health Outcomes Framework (PHOF). We will be downloading data from source and the exercise is to test the relationship between diabetes care processes and outcome.

To do this we will be performing supervised analysis of diabetes data from the PHOF - constructing linear models with diabetes outcome as the dependent variable and care processes and the independent (predictor) variables. To allow for diabetes frequency we will also include diabetes prevalence in our models, as well as a summary measure of socio-economic status (SES). In England, the Index of Multiple Deprivation 2019 (IMD) is widely used as a summary SES index.

We will be analysing the data for sub-national health administrative units called sub-ICBs (SICB). England is subdivided into 104 SICBs - these are the units of health care planning and performance.

## Get started

First we need to load the R packages we need to extract data and for analysis.

needs(tidyverse, pak, tidymodels, mgcv, glmnet, corrplot, factoextra, FactoMineR, umap, dbscan, broom, cluster, fpc)

## Get the data

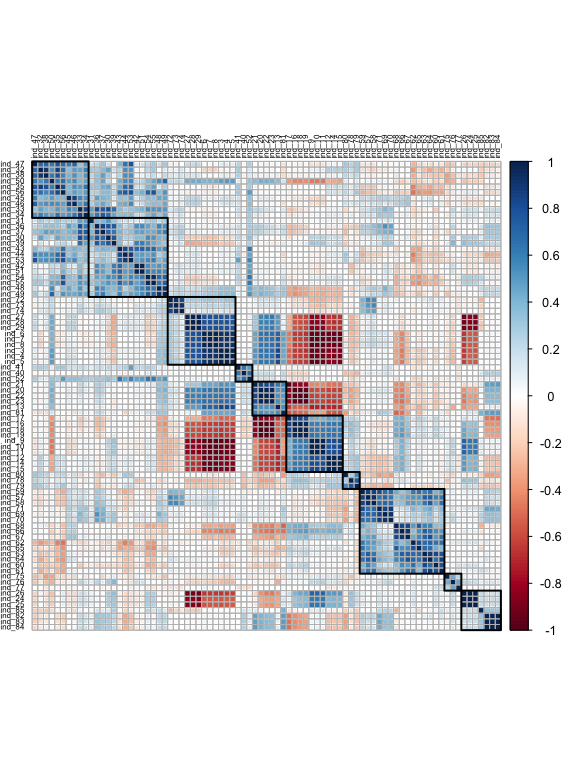
Now we load the diabetes data. he code segment below shows how data is loaded.

diabetes\_model <- read\_csv("https://github.com/julianflowers/spha/blob/main/dm\_model\_data.csv?raw=TRUE", show\_col\_types = FALSE)  
diabetes\_codebook <- read\_csv("https://github.com/julianflowers/spha/blob/main/dmmod\_codebook.csv?raw=TRUE", show\_col\_types = FALSE)

The resulting dataset has 104 records, and consists of 84 diabetes metrics.

## Explore the data

The first step is explore relationships between variables. A common way to do this is to construct a correlation matrix or a correlogram. In R, the corrplot package is widely used for this.

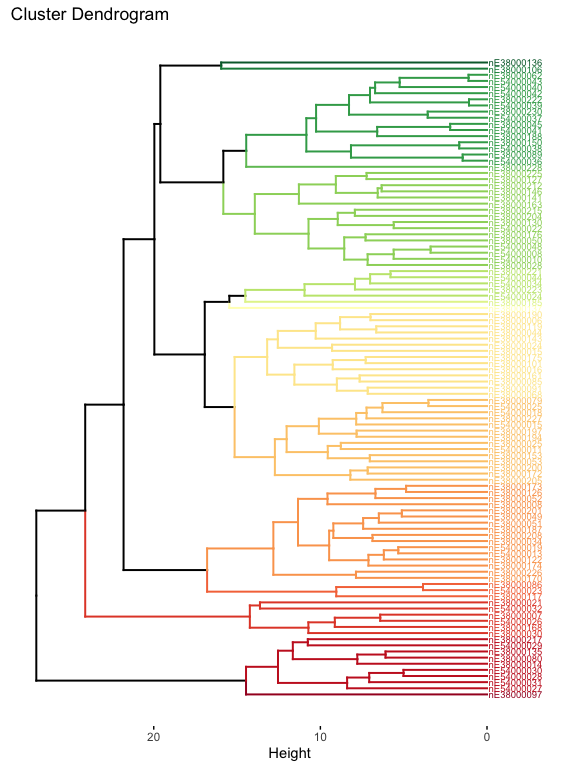


## Cluster analysis

The next step is to identify patterns in the data. We’ll start with cluster analysis - this has two main forms - hierarchical and non-hierarchical. Hierarchical clustering creates a tree diagram (dendrogram)

### Hierarchical clustering

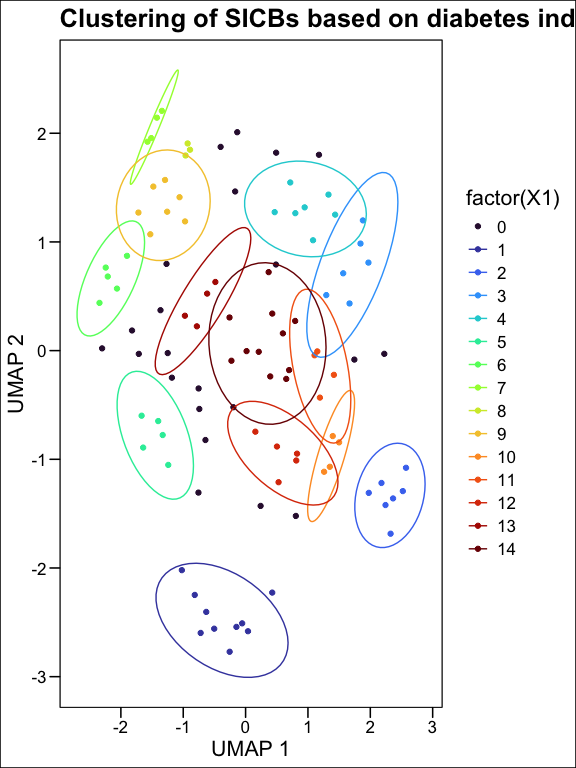
The groups areas which have similar patterns of indicator values



## Clustering algorithms

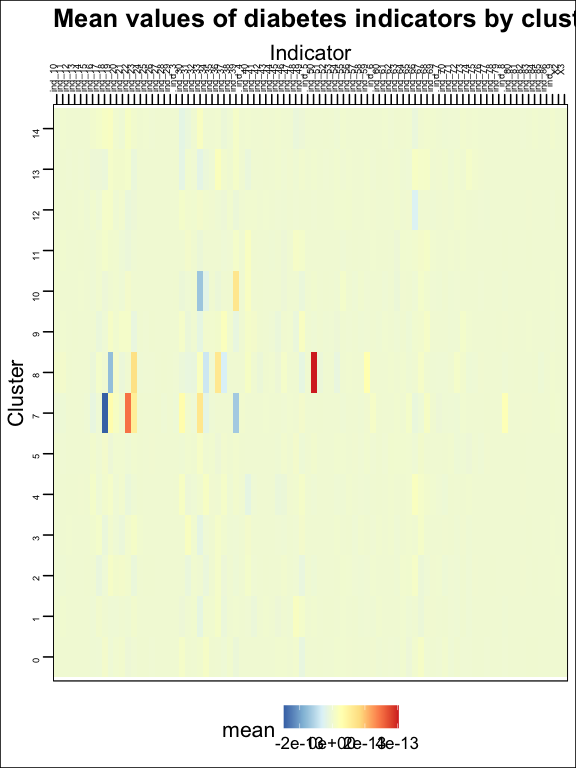
A recent approach to identifying patterns in large multivariate datasets is to take a 2 step approach:

1. Dimensionality reduction - first collapse the data to 2 dimensions. There are several ways to do this including principal component analysis (PCA), Uniform Manifold Approximation and Projection (UMAP)(Hozumi et al. 2021; Konopka 2023). UMAP is increasingly used because it preserves small scale structure in the data and can emphasize clustering.
2. Then apply a clustering algorithm. DBSCAN is often used (Hahsler and Piekenbrock (2023)). This doesn’t require pre-specified cluster number, but it is based on a nearest neighbour algorithm. The choice of nearest neighbour count (minPts) determines the number of clusters found by the algorithm.



## Cluster features

Next, we can review how the clusters differ in terms of diabetes indicators.



## Modelling diabetes control and outcome

### Population diabetes outcomes

First we’ll explore the relationship between diabetes outcome and diabetes control. We will use minor amputation rates as an outcome measure (Lower is better), and diabetes control metrics as predictors We’ll use the most recent values of predictor variables (2020). The latest year for amputation rates is 2018.

#### Linear model

| **Characteristic** | **Beta** | **95% CI**1 | **p-value** |
| --- | --- | --- | --- |
| ind\_59 | -0.69 | -1.5, 0.10 | 0.087 |
| ind\_62 | 0.45 | 0.17, 0.74 | 0.002 |
| ind\_65 | 0.01 | -0.58, 0.61 | >0.9 |
| ind\_68 | -0.42 | -1.0, 0.18 | 0.2 |
| ind\_71 | 0.41 | 0.01, 0.81 | 0.042 |
| ind\_74 | 0.21 | -0.17, 0.59 | 0.3 |
| ind\_77 | 0.00 | -0.06, 0.05 | >0.9 |
| ind\_80 | 0.01 | -0.06, 0.09 | 0.8 |
| ind\_84 | 0.33 | -0.21, 0.87 | 0.2 |
| ind\_81 | 0.03 | -0.04, 0.10 | 0.4 |
| 1CI = Confidence Interval | | | |
| R² = 0.326; Adjusted R² = 0.253; Sigma = 1.81; Statistic = 4.49; p-value = <0.001; df = 10; Log-likelihood = -203; AIC = 431; BIC = 462; Deviance = 304; Residual df = 93; No. Obs. = 104 | | | |

Simple linear regression suggests that the proportion of the population achieving all 3 treatment targets is inversely associated with lower minor amputation rates. It is positively associated with tight blood pressure control and tight diabetes control. The model explains 32.56% of the variance in minor amputation rates. The root mean squared error is 1.71.

#### Non-linear model

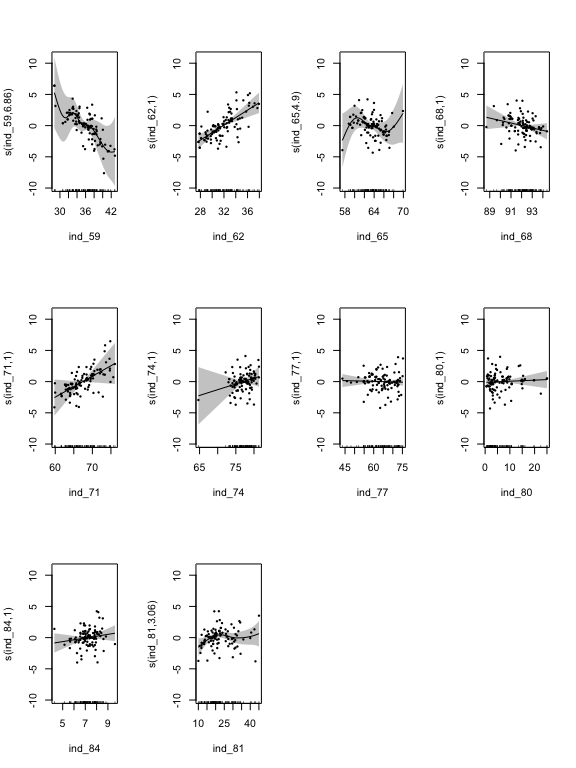
We can extend our inferential model to include non-linear relationships. We can use the mgcv package to fit a generalised additive model (GAM) to the data. This allows us to model non-linear relationships between the dependent and independent variables.

| **Characteristic** | **Beta** | **95% CI**1 | **p-value** |
| --- | --- | --- | --- |
| s(ind\_59) |  |  | 0.042 |
| s(ind\_62) |  |  | <0.001 |
| s(ind\_65) |  |  | 0.079 |
| s(ind\_68) |  |  | 0.2 |
| s(ind\_71) |  |  | 0.081 |
| s(ind\_74) |  |  | 0.3 |
| s(ind\_77) |  |  | 0.7 |
| s(ind\_80) |  |  | 0.6 |
| s(ind\_84) |  |  | 0.3 |
| s(ind\_81) |  |  | 0.2 |
| 1CI = Confidence Interval | | | |
| df = 23; Log-likelihood = -183; AIC = 413; BIC = 476; Deviance = 204; Residual df = 81; No. Obs. = 104 | | | |

##   
## Family: gaussian   
## Link function: identity   
##   
## Formula:  
## ind\_85 ~ s(ind\_59) + s(ind\_62) + s(ind\_65) + s(ind\_68) + s(ind\_71) +   
## s(ind\_74) + s(ind\_77) + s(ind\_80) + s(ind\_84) + s(ind\_81)  
##   
## Parametric coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 7.962 0.155 51.2 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Approximate significance of smooth terms:  
## edf Ref.df F p-value   
## s(ind\_59) 6.86 7.90 2.19 0.042 \*   
## s(ind\_62) 1.00 1.00 19.75 2.8e-05 \*\*\*  
## s(ind\_65) 4.90 6.03 1.97 0.079 .   
## s(ind\_68) 1.00 1.00 1.98 0.163   
## s(ind\_71) 1.00 1.00 3.13 0.081 .   
## s(ind\_74) 1.00 1.00 0.98 0.326   
## s(ind\_77) 1.00 1.00 0.11 0.742   
## s(ind\_80) 1.00 1.00 0.22 0.641   
## s(ind\_84) 1.00 1.00 1.23 0.271   
## s(ind\_81) 3.06 3.79 1.72 0.151   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.427 Deviance explained = 54.8%  
## GCV = 3.2167 Scale est. = 2.5108 n = 104

This shows 4 variables, blood pressure control, diabetes control, cholesterol control and HbA1c control are non-linearly associated with minor amputation rates. The model explains 55% of the variance in minor amputation rates. The root mean squared error is 1.4 - the model is a better fit than the linear version

We can plot the relationship between the predictors and the outcome to visualise the non-linear relationships.



Plot of GAM model of diabetes control and minor amputation rates

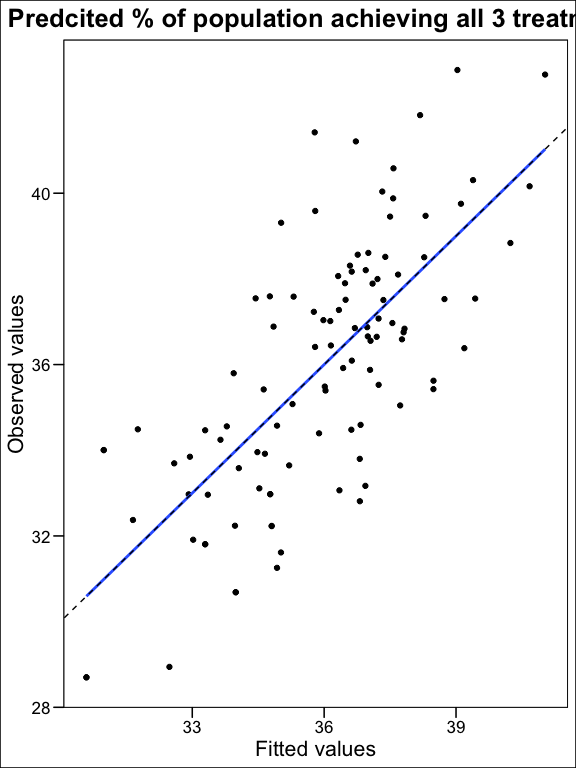
This confirms the lack of relationship with ind\_77, ind\_80 and ind\_65, inverse linear relationships with ind\_59 and ind\_68, and positive relationship with ind\_62 and ind\_71. The other predictors show a weak non-linear relationship with minor amputation rates.

Decoding these relationships, we learn that the proportion of the diabetic population achieving treatment targets is inversely associated with minor amputation rates - this is encouraging as it suggests that better control of diabetes is associated with lower rates of minor amputations. The relationship is linear, with the lowest minor amputation rates seen at the highest levels of diabetes control, and suggests a target for further reduction in amputation rates by increasing the percentage controlled (currently below 44%), and reducing the variation between areas.

Somewhat paradoxically, individual treatment targets - ind\_62 (cholesterol control) and ind\_71 (blood pressure control) are positively associated with minor amputation rates, suggesting that

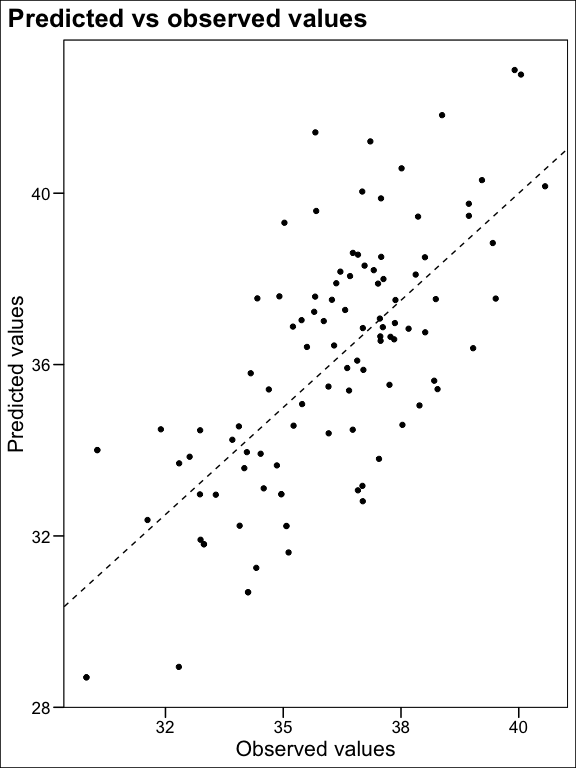
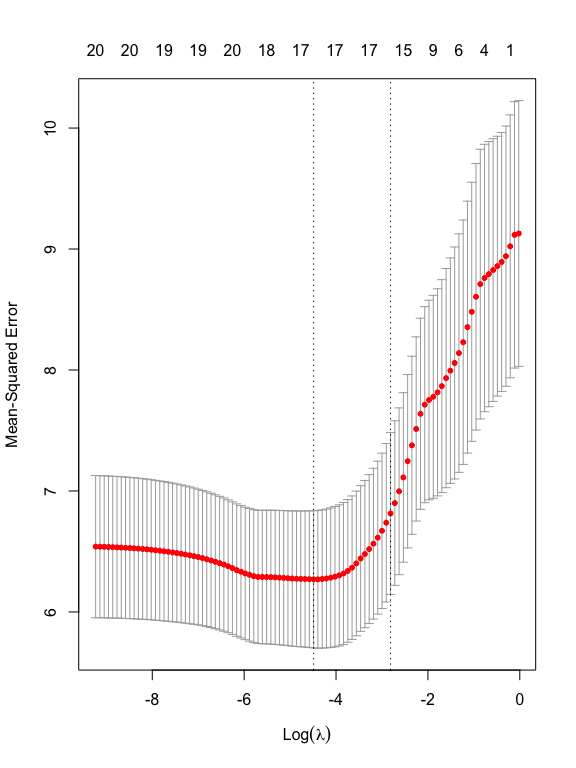
The proportion of the population achieving all 3 treatment targets is a measure of population treatment control. We’ll use the most recent value (ind\_59) as the dependent variable, and the most recent values of predictor variables (2020).

To further explore the relationship between achieving all 3 treatment targets and the predictors we can use a linear model. Initially we will fit all predictors (process, demography and prevalence estimates).



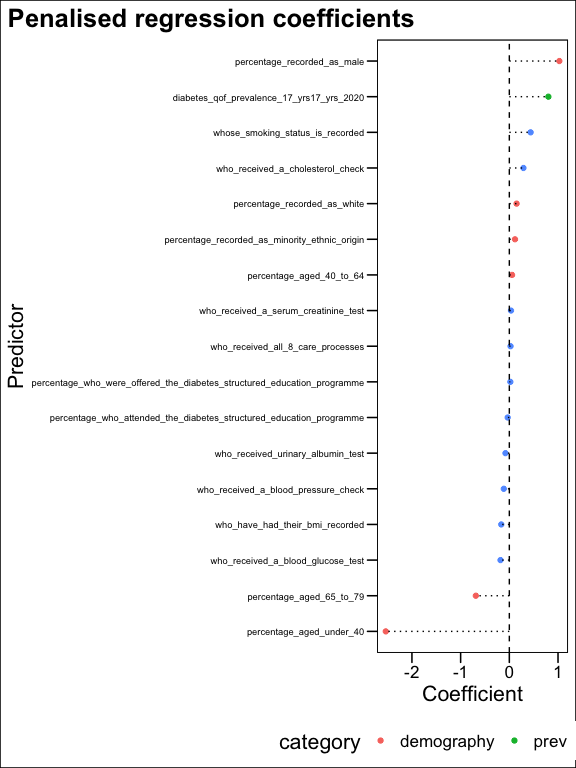
The chart shows a linear model is a moderate fit with an r squared value is 0.52. There is residual variation - the root mean squared error is 2.07 is 2.1. This means our model predicts the proportion of the population achieving all 3 treatment targets within 2.1% of the observed value.

With a large number of predictors, model selection (i.e. best fit choice of predictor variables) is important. We can use the step function to select the best model i.e. performance of the model with the fewest predictors. However, penalised regression is a more efficient technique. This automatically selects the best model by penalising the number of predictors. In R the glmnet package is widely used for this (Friedman, Tibshirani, and Hastie (2010)). It uses lasso regression to remove variables which contribute least to the model.



The error of this model is 2.09 - similar to the linear model

## Plot regression coefficients



Demographic, prevalence and care process predictors of population diabetes control

## Annex

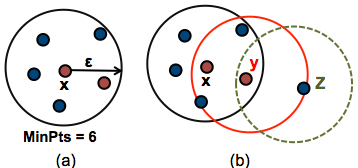
**Algorithm of DBSCAN**

The goal is to identify dense regions, which can be measured by the number of objects close to a given point.

Two important parameters are required for **DBSCAN**: **epsilon** (“eps”) and **minimum points** (“MinPts”). The parameter **eps** defines the radius of neighbourhood around a point x. It’s called called the -neighbourhood of x. The parameter **MinPts** is the minimum number of neighbours within “eps” radius.

Any point x in the dataset, with a neighbour count greater than or equal to **MinPts**, is marked as a **core point**. We say that x is **border point**, if the number of its neighbours is less than MinPts, but it belongs to the -neighbourhood of some core point z. Finally, if a point is neither a core nor a border point, then it is called a noise point or an outlier.

The figure below shows the different types of points (core, border and outlier points) using **MinPts = 6**. Here x is a core point because , y is a border point because , but it belongs to the -neighbourhood of the core point x. Finally, z is a noise point.



Density based clustering basic idea - minimal point and epsilon

We define 3 terms, required for understanding the **DBSCAN algorithm**:

* **Direct density reachable**: A point “A” is **directly density reachable** from another point “B” if: i) “A” is in the -neighbourhood of “B” and ii) “B” is a core point.
* **Density reachable**: A point “A” is **density reachable** from “B” if there are a set of core points leading from “B” to “A.
* **Density connected**: Two points “A” and “B” are **density connected** if there are a core point “C”, such that both “A” and “B” are **density reachable** from “C”.

A **density-based cluster** is defined as a group of density connected points. The algorithm of density-based clustering (DBSCAN) works as follow:

The algorithm of density-based clustering works as follow:

1. For each point , compute the distance between and the other points. Finds all neighbour points within distance **eps** of the starting point (). Each point, with a neighbour count greater than or equal to **MinPts**, is marked as **core point** or **visited**.
2. For each **core point**, if it’s not already assigned to a cluster, create a new cluster. Find recursively all its density connected points and assign them to the same cluster as the core point.
3. Iterate through the remaining unvisited points in the dataset.

Those points that do not belong to any cluster are treated as outliers or noise

Friedman, Jerome, Robert Tibshirani, and Trevor Hastie. 2010. “Regularization Paths for Generalized Linear Models via Coordinate Descent” 33. <https://doi.org/10.18637/jss.v033.i01>.

Hahsler, Michael, and Matthew Piekenbrock. 2023. “Dbscan: Density-Based Spatial Clustering of Applications with Noise (DBSCAN) and Related Algorithms.” <https://CRAN.R-project.org/package=dbscan>.

Hozumi, Yuta, Rui Wang, Changchuan Yin, and Guo-Wei Wei. 2021. “UMAP-Assisted K-Means Clustering of Large-Scale SARS-CoV-2 Mutation Datasets.” *Computers in Biology and Medicine* 131 (April): 104264. <https://doi.org/10.1016/j.compbiomed.2021.104264>.

Konopka, Tomasz. 2023. “Umap: Uniform Manifold Approximation and Projection.” <https://CRAN.R-project.org/package=umap>.