corVis: An R Package for Visualising Associations and Conditional Associations

by Amit Chinwan and Catherine Hurley

Abstract Correlation matrix displays are important tools to explore multivariate datasets. These displays with other measures of association can summarize interesting patterns to an analyst and assist them in framing questions while performing exploratory data analysis. In this paper, we present new visualisation techniques to visualise association between all the variable pairs in a dataset in a single plot, which is something existing displays lack. Also, we propse new methods to visualise relationship among variable pairs using conditioning. We use different layouts like matrix or linear for our displays. We use seriation in our displays which helps in highlighting interesting patterns easily. The R package `corVis` provides an implementation.

`See comments on abstract made on Dec 21, when I requested re-write.`

Section 1: Introduction

Correlation matrix display helps in visually exploring correlations among variables during Exploratory Data Analysis (EDA) of a multivariate dataset. Popularized by Friendly (2002) as corrgram, these displays are produced by first calculating the correlation among the variables and then plotting these calculated values in a matrix display. With effective ordering techniques, these displays quickly highlight variables which are highly correlated and an analyst interested in building a predictive model could use these displays to flag and avoid multicollinearity.

The correlation displays are generally used with Pearson's correlation coefficient and are therefore limited to quantitative variables. An analyst can use one-hot encoding of the qualitative variables in order to use these displays but will need to deal with the high dimensions as a result of the encoding. In addition to the dimensionality problem, it is not easy to assess the overall correlation when using the one-hot encoding.

Tukey and Tukey (Tukey and Tukey, 1985) introduced scagnostics which are diagnostic measures, quantifying density, shape, trend, and other features, of a scatterplot. Along with scagnostics, they proposed a scagnostics scatterplot matrix which is a visual display to explore and compare these measures for all the variable pairs in a dataset. By comparing multiple measures at once, variable pairs with high difference among measures could be identified and looked at in more detail. In a similar manner, a display comparing association measures will help in finding interesting variable pairs. Many association measures have been proposed to summarize different types of relationships. The most commonly used measure is Pearson's correlation coefficient which captures any linear trend present between the variables. Other popular measures include Kendall's or Spearman's rank correlation coefficient which are non-parametric measures and look for monotonic relationship. Distance correlation (Székely et al., 2007) is an important measure useful in exploring non-linear relationships. The information theory measure maximal information coefficient (MIC) (Reshef et al., 2011) is capable of summarizing complex relationships. With an effective displaying technique for the multiple measures of association, it will be easier to .

Small multiples (or Trellis display) is a simple yet powerful approach to compare partitions of data and understand multidimensional datasets (Tufte, 1986). The display is produced by splitting the data into groups by a conditioning variable and then plotting the data for each group. Such displays allow analysts to quickly infer about the impact of the conditioning variable. A similar idea applied to displays of association measures (correlation plot) will help uncover underlying patterns in the data. One such pattern is Simpson's paradox which can be detected by comparing Pearson's correlation for data at overall level versus individual levels of the conditioning variable.

In this paper, we propose extensions of the correlation plot and new visualizations which look at variables of mixed type, multiple association measures and conditional associations. These displays are implemented in the R package **corVis**. The next section provides a review of existing packages which deal with correlation displays and a quick background on association measures and the packages used for calculating them. Then we describe our approach to calculate the association measures, followed by visualizations of associations and conditional associations. We conclude with a summary and future work.

Section 2: Background

In this section we provide a brief review of existing packages used for correlation displays and association measure calculation.

Section 2.1: Literature Review on Correlation Displays

According to Hills (1969), "the first and sometimes only impression gained by looking at a large correlation matrix is its largeness". To overcome this, Murdoch and Chow (1996) proposed a display for large correlation matrices which uses a matrix layout of ellipses where the parameters of the ellipses are scaled to the correlation values. Friendly (2002) expanded on this idea by rendering correlation values as shaded squares, bars, ellipses, or circular 'pac-man' symbols. The variables in the matrix displays were optionally ordered using the angular ordering of the first two eigen vectors of the correlation matrix. The ordering places highly-correlated pairs of variables nearby, making it easier to quickly identify groups of variables with high mutual correlation.

Nowadays, there are many R packages devoted to correlation visualisation. Table 1 provides a summary, listing the displays offered, and whether these extend to factor variables or mixed numeric-factor pairs.

The R package **corrplot** (Wei and Simko, 2021) provides an implementation of the methods in Friendly (2002). The package **corr** (Kuhn et al., 2020) organises correlations as tidy data, so leveraging the data manipulation and visualisation tools of the **tidyverse** (Wickham et al., 2019). In addition to various matrix displays, the package offers network displays where line-thickness encodes correlation magnitude, with a filtering option to discard low-correlation edges.

The package **corrgrapher** (Morgen and Biecek, 2020) uses a network plot for exploring correlations, where the nodes close to each other have high correlation magnitude, edge thickness encodes the absolute correlation value and edge color indicates the sign of correlation. The package also handles mixed type variables by using association measures obtained as transformations of p-values obtained from Pearson's correlation test in the case of two numeric variables, Kruskal's test for numerical and factor variables, and a chi-squared test for two categorical variables.

The package **linkspotter** (Samba, 2020) offers a variety of association measures (distance correlation, MIC, maximum normalized mutual information) in addition to correlation, where the measure used depends on whether the variables are both numerical, categorical or mixed. The results are visualized in a network plot, which may be packaged into an interactive shiny application.

Our own package **corVis** offers a variety of displays, and has new features not available elsewhere, in particular simultaneous display of multiple association measures, and association displays stratified by levels of a grouping variable. This will be described in the following sections.

There have been other extensions to correlation displays which are useful when dealing with high dimensional datasets. Hills (1969) proposed a QQ plot of the z-transform of the entries of the correlation matrix to discover correlation coefficients too large to come from a normal distribution with mean zero. Buja et al. (2016) proposed Association Navigator which is an interactive visualization tool for large correlation matrices with upto 2000 variables. The R package **scorrplot** (Gerber, 2022) produces an interactive scatterplot for exploring pairwise correlations in a large dataset by projecting variables as points on a scatterplot with respect to some user-selected variables of interest, driven by a geometric interpretation of correlation and encoding the correlation as vertical gridlines in the plot. The package allows user to update variable of interest which creates tour of the correlation space between different projections of the data.

The R package **correlationfunnel** offers a novel display which assists in feature selection in a setting with a single response and many predictor variables. All numeric variables including the response are binned. All (now categorical) variables in the resulting dataset are one-hot encoded and Pearson's correlation calculated with the response categories. The correlations are visualised in a dot-plot display, where predictors are ordered by maximum correlation magnitude. Correlations between one-hot encoded variables are challenging to interpret, especially as the number of levels increase. In corVis we offer a similar dot-plot display, but showing multiple correlation or association measures, or alternatively measures stratified by a grouping variable.

Section 2.2: Literature Review on Association Measures

An association measure is defined as a numerical summary quantifying the relationship between two or more variables. The measure is called symmetric if its value is invariant to the choice of independent or dependent variable during the calculation. For example, Pearson's correlation coefficient summarizes the strength and direction in the range $[-1, 1]$ of the linear relationship present between two numeric

Table 1: List of the R packages dealing with correlation or correlation displays with information on whether the plots display multiple measures, conditional display of measures and mixed variables in a single plot

Package	Display	MixedVariables
corrplot	heatmap	
corrr	heatmap/network	
corrgrapher	network	
linkspotter	network	Yes
correlation	heatmap/network	
corVis	heatmap/matrix/linear	Yes

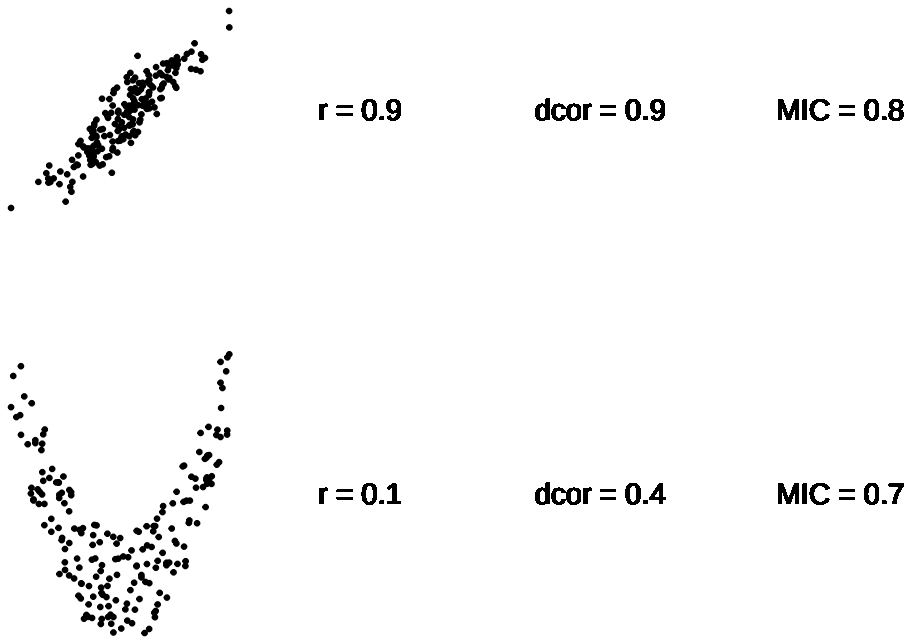

Figure 1: Multiple association measures for simulated linear and non-linear pattern. The first row of the plot shows a linear pattern, the value of Pearson's correlation, distance correlation and MIC. The second row of the plot shows a non-linear pattern along with the values for association measures. All three association measures show a high value for the linear relationship and hence are useful for linear patterns. For the non-linear pattern, distance correlation and MIC are more suitable measures than Pearson's correlation.

variables and is symmetric. Kendall's or Spearman's rank correlation coefficient are other popular measures which assess monotonic relationship in interval $[-1, 1]$ among two numeric variables and are symmetric measures.

Pearson's correlation is the most popular association measure because of its easier calculation and interpretability but its limitations such as influence of outliers on its value and measuring only linear dependencies makes it a less useful measure of association overall. The recent measures such as distance correlation (Székely et al., 2007) and MIC (Reshef et al., 2011) overcome these limitations and are more suitable for datasets with both linear and non-linear patterns.

Figure 1 shows a plot of simulated linear and non-linear patterns. The first row shows a linear relationship along with the value for measures such as Pearson's correlation, distance correlation and maximal information coefficient respectively for the pattern. In a similar manner, the second row shows a quadratic relationship and its values for Pearson's correlation, distance correlation and maximal information coefficient respectively.
It is clearly evident from Figure 1 that all the three measures summarizes the pattern with linear relationship quite well. For the non-linear pattern, distance correlation and MIC are better in detecting underlying relationship than Pearson's correlation. This suggests that association measures such as distance correlation, MIC and other measures should be used along with Pearson's correlation for exploring relationships among variables in the datasets where there is no prior knowledge about the possible patterns in the data.

The distance correlation coefficient (Székely et al., 2007) is an association measure which looks for any relationship among two numeric variables using the distances between observations of these variables and summarizes the relationship in $[0, 1]$. The distance correlation is 0 only when the variables are independent and is a symmetric measure.

The maximal information coefficient (MIC) (Reshef et al., 2011) is an information theory measure which uses mutual information among the two variables for its calculation. The main idea is to find a grid out of possible grids on a scatterplot of two numeric variables, in order to discretize the variables, which maximises the mutual information for the two variables. A normalisation technique is used to make the mutual information from different grids comparable. Referred as 'a correlation of 21st century' (Speed, 2011), MIC is capable of summarizing different types of relationships, not just linear or monotonous, between numeric variables and is in range $[0, 1]$. Reshef et al. (2011) used MIC and other related statistics to explore pairwise relationships in large data sets such as major-league baseball, gene expression, global health, and the human gut microbiota.

Both distance correlation and MIC have advantages of detecting non-linear and complex relationship but these measures aren't perfect yet. Simon and Tibshirani (2014) showed that distance correlation has more statistical power than MIC. Also, distance correlation is not an approximation when compared to MIC. On the other hand, distance correlation computation is slower as compared to the conventional association measures such as Pearson's correlation for datasets with high number of cases.

In addition to association measures for numeric variables, association measures for ordinal, nominal and mixed variable pairs are useful in exploring a multivariate dataset. We now give an overview of available association measures for other variable types.

Agresti (2010) provides an overview of the association measures which are used for exploring association between ordinal variables. Kendall's tau-b (Kendall, 1945) is an association measure useful in summarizing the relationship in range $[-1, 1]$ between two ordinal variables. It is a relatively stable measure than Goodman and Kruskal's gamma with respect to the changes in categories of any variable i.e. if two categories are merged to make a single category. The polychoric correlation (Olsson, 1979) measures the correlation between two ordinal variables by assuming two normally distributed latent variables for a contingency table of two ordinal variables and summarizes the association in $[-1, 1]$.

The association measures for the case of nominal pair of variables should be invariant to the order in which the categories appear. Pearson's contingency coefficient uses the χ^2 value from the Pearson's χ^2 test for independence and is a useful measure to summarize the association in $[0, 1]$ between two nominal variables. Another measure for nominal variable pair is the Uncertainty coefficient (Theil, 1970) measuring the proportion of uncertainty in one variable which is explained by the other variable. The uncertainty coefficient measure is in the range $[0, 1]$ and is not symmetric.

Section 3: Introducing corVis

corVis is an R package which calculates measures of association for every variable pair in a dataset and provides visualizations for displaying associations. Most of the existing correlation displays are limited to numeric pairs of variables. This package extends these displays to every variable pair. The main goal of our work is to propose displays for multiple association measures and conditional associations display which are useful for uncovering interesting patterns in the data. This will help in identifying variable pairs which shows a type of relationship or pattern in a dataset with large number of variables.

While designing these displays we consider matrix and linear layouts. A matrix layout reduces the effort in looking up for variable pairs corresponding to a cell or panel, and different measures may be displayed on the upper and lower triangle of the matrix. On the other hand, the filtering of variable pairs, for example pairs having measure value greater than a threshold, is easier with linear layouts in comparison to matrix layouts.

Table 2 provides a list of the functions available in the package. The functions `calc_assoc` and `calc_assoc_all` are responsible for calculating association measures which are used as input for the `plot_assoc_matrix` and `plot_assoc_linear` functions. The functions `plot_assoc_matrix` and `plot_assoc_linear` produces association display, multiple association measures display and conditional association display, in a matrix and linear layout respectively. We provide detailed examples on calculation and visualisation of association and conditional association in next sections.

Table 2: List of functions in corVis package

Function	Usage	Description
calc_assoc	Calculation	Calculates association measures
calc_assoc_all	Calculation	Calculates all the association measures available in package
plot_assoc_matrix	Visualization	Visualize assocition and conditional association in matrix plot
plot_assoc_linear	Visualization	Visualize assocition and conditional association in linear plot
show_assoc	Visualization	Association (or conditional) plot for a pair of variables

Section 4: corVis: Calculating Association

This section describes the calculation of association measures in our package **corVis**. The package provides a standard interface for calculating a collection of various measures of association which quantifies the relationship between two variables. The association measures available in the package are not limited to numeric variables and are used with nominal, ordinal and mixed variable pairs as well. The package also provides a functionality for handling missing value or `NA` while calculating the association measures.

Table 3 lists different functions provided in the package to calculate varoius measures of association. The `Function` column represents `tbl_*` functions which are used to calculate a single association measure. The `typeX` and `typeY` columns provide the information on types of variables which can be used with the corresponding functions. The `X` or `Y` variable is one of the numeric, nominal or ordinal type. The `from` column corresponds to the external package functions used to calculate the association measures by `tbl_*` functions. The `symmetric` column represents if the measure is symmetric i.e. if its value doesn't change by the choice of independent or dependent variable during its calculation. The last column provides the range of values for these measures. The function `tbl_easy` calculates association measures available in the R package **correlation** and is suitable for different variable types.

The functions in Table 3 such as `tbl_ace`, `tbl_cancor` and `tbl_nmi` which calculates maximal correlation coefficient, canonical correlationa and normailzed mutual information respectively, have been implemented in **corVis**.

For numeric pairs of variables, the package provides a range of association measures. The popular correlation coefficients like Pearson's, Spearman's or Kendall's are calculated using `tbl_cor` function. The measures such as distance correlation or MIC are calculated using `tbl_dcor` or `tbl_mine` respectively. The association measures available in the package for the ordinal pairs of variables are polychoric correlation and Kendall's coefficients which are calculated using `tbl_polycor` or `tbl_tau` respectively. For nominal pairs of variables, the functions like `tbl_gkTau`, `tbl_gkGamma`, `tbl_uncertainty`, `tbl_chi`, `tbl_cancor` are used for exploring association among the variables.

The function `tbl_cancor` calculates a measure of association based on canonical correlations for mixed pairs of variables. Nominal variables are converted into sets of dummy variables, which are then assigned score to find the maximal correlation. For two numeric variables this measure is identical to absolute correlation, for two factors the correlation is identical to that obtained from correspondence analysis.

The functions listed in Table 3 for calculating association measures provide a functionality for handling missing value or `NA` in the dataset. Each of these functions either have a `handle.na` argument or automatically uses pairwise complete observations (depending on the package used for calculation) for taking care of missing values present in the data.

Calculating association for a single type of variable pairs

We have a function which creates a tibble structure for the variable pairs in a dataset along with calculated association measure. The package contains various functions (shown in Table 3) for different association measures in the form `tbl_*` to calculate them. For example, in order to calculate distance correlation for numeric pair of variables in a dataset, the function `tbl_dcor` is used.

```
distance <- tbl_dcor(iris)
distance
```

```
#> # A tibble: 6 x 4
#>   x            y            measure measure_type
#>   <chr>        <chr>          <dbl> <chr>
#> 1 Sepal.Width  Sepal.Length   0.311 dcor
```

Table 3: List of the functions available in the package for calculating different association measures along with the packages used for calculation.

Function	X	Y	from	symmetric	range
tbl_cor	numerical	numerical	stats::cor	Y	[-1,1]
tbl_dcor	numerical	numerical	energy::dcor2d	Y	[0,1]
tbl_mine	numerical	numerical	minerva::mine	Y	[0,1]
tbl_ace	numerical	numerical	corVis	Y	[0,1]
tbl_polycor	ordinal	ordinal	polycor::polychor	Y	[-1,1]
tbl_tau	ordinal	ordinal	DescTools::KendalTauA,B,C,W	Y	[-1,1]
tbl_gkGamma	ordinal	ordinal	DescTools::GoodmanKruskalGamma	Y	[-1,1]
tbl_gkTau	nominal	nominal	DescTools::GoodmanKruskalTau	N	[0,1]
tbl_uncertainty	nominal	nominal	DescTools::UncertCoef	Y	[0,1]
tbl_chi	nominal	nominal	DescTools::ContCoef	Y	[0,1]
tbl_cancor	nominal/numerical	nominal/numerical	corVis	Y	[0,1]
tbl_nmi	nominal	nominal	corVis	Y	[0,1]
tbl_easy	nominal/numerical	nominal/numerical	correlation::correlation	Y	[-1,1]

```
#> 2 Petal.Length Sepal.Length   0.859 dcor
#> 3 Petal.Width  Sepal.Length   0.827 dcor
#> 4 Petal.Length Sepal.Width    0.542 dcor
#> 5 Petal.Width  Sepal.Width    0.513 dcor
#> 6 Petal.Width  Petal.Length   0.974 dcor
```

In the tibble output for the functions mentioned in Table 3 `x` and `y` represents a pair of variables. The `measure` variable represents the calculated value for association measure. And the `measure_type` variable represents the association measure calculated for `x` and `y` pair.

Calculating association measures for whole dataset

`calc_assoc` is used to calculate association measures for all variable pairs in a dataset at once in a tibble structure. The variable pairs in the output are unique pairs in the dataset where $x \neq y$. Because of the tidy structure of the output, the data manipulation and visualisation tools of **tidyverse** (Wickham et al., 2019) are applicable and are useful for further exploration of pairwise associations. In addition to tibble structure, the output also has `pairwise` and `data.frame` class which are important class attributes for producing visual summaries in this package.

The function `calc_assoc` has a `types` argument which is a tibble of the `tbl_*` functions for different types of variable pairs. The default tibble is `default_assoc()` which includes `tbl_cor` if both the variables are numeric and calculates Pearson's correlation , `tbl_gkGamma` if both the variables are ordinal and calculates Goodman and Kruskal's gamma , `tbl_cancor` if one is factor and other is numeric and calculates canonical correlation, and canonical correlation for the rest of the variable pairs.

```
default_measures <- default_assoc()
default_measures
```

```
#> # A tibble: 4 x 4
#>   funName     typeX   typeY   argList
#>   <chr>       <chr>   <chr>   <list>
#> 1 tbl_cor     numeric numeric <NULL>
#> 2 tbl_gkGamma ordered ordered <NULL>
#> 3 tbl_cancor  factor  numeric <NULL>
#> 4 tbl_cancor  other   other   <NULL>
```

```
iris_assoc <- calc_assoc(d = iris,
                         types = default_measures)
iris_assoc
```

```
#> # A tibble: 10 x 4
#>    x            y            measure measure_type
#>    <chr>        <chr>          <dbl> <chr>
#>  1 Sepal.Width  Sepal.Length  -0.118 pearson
#>  2 Petal.Length Sepal.Length   0.872 pearson
#>  3 Petal.Width  Sepal.Length   0.818 pearson
#>  4 Species      Sepal.Length   0.787 cancor
```

```
#>  5 Petal.Length Sepal.Width   -0.428 pearson
#>  6 Petal.Width  Sepal.Width   -0.366 pearson
#>  7 Species      Sepal.Width    0.633 cancor
#>  8 Petal.Width  Petal.Length   0.963 pearson
#>  9 Species      Petal.Length   0.970 cancor
#> 10 Species      Petal.Width    0.964 cancor

class(iris_assoc)

#> [1] "pairwise"   "tbl_df"     "tbl"        "data.frame"
```

The default tibble of measures is updated using the `update_assoc` function which has arguments for updating the `tbl_*` functions to calculate association measures depending on the type variable pair in the dataset and a method for `tbl_*` functions which calculates more than one measure. The `update_assoc` function has an argument `default` which has the `default_assoc()` tibble as its default value and is useful when `tbl_*` functions need to be updated for a few types of variable pairs.

```
updated_assoc <- update_assoc(default_measures,
                              num_pair = "tbl_cor",
                              num_pair_argList = "spearman",
                              mixed_pair = "tbl_cancor",
                              other_pair = "tbl_nmi")
updated_assoc

#> # A tibble: 4 x 4
#>   funName     typeX   typeY   argList
#>   <chr>       <chr>   <chr>   <list>
#> 1 tbl_cor     numeric numeric <chr [1]>
#> 2 tbl_gkGamma ordered ordered <NULL>
#> 3 tbl_cancor  factor  numeric <NULL>
#> 4 tbl_nmi     other   other   <NULL>

updated_iris_assoc <- calc_assoc(d = df,
                                 types = updated_assoc)
updated_iris_assoc

#> # A tibble: 3 x 4
#>   x           y        measure measure_type
#>   <chr>       <chr>      <dbl> <chr>
#> 1 Usage       Function   0.647 nmi              this looks wrong
#> 2 Description Function   1     nmi
#> 3 Description Usage      0.647 nmi
```

`calc_assoc` also has a `handle.na` argument for handling the `NA` or missing values which is fed into the `tbl_*` functions used with the `types` argument for different types of variable pairs. The default value is set to `TRUE` for using pairwise complete observations for calculating a measure of association between two variables.

Calculating multiple association measures

The multiple association measures are calculated using `calc_assoc_all` function in the package. The function takes a dataset and a list of measures as input and outputs a tibble structure with multiple measures of association for every variable pair. This output serves as input to the multiple association measures plot function for comparison of measures for variable pairs.

```
#> # A tibble: 22 x 4
#>    x            y            measure measure_type
#>    <chr>        <chr>          <dbl> <chr>
#>  1 Sepal.Width  Sepal.Length  -0.118 pearson
#>  2 Petal.Length Sepal.Length   0.872 pearson
#>  3 Petal.Width  Sepal.Length   0.818 pearson
#>  4 Petal.Length Sepal.Width   -0.428 pearson
#>  5 Petal.Width  Sepal.Width   -0.366 pearson
#>  6 Petal.Width  Petal.Length   0.963 pearson
#>  7 Sepal.Width  Sepal.Length   0.118 cancor
```

```
#>  8 Petal.Length Sepal.Length   0.872 cancor
#>  9 Petal.Width  Sepal.Length   0.818 cancor
#> 10 Species      Sepal.Length   0.787 cancor
#> # ... with 12 more rows
```

Calculating conditional association

`calc_assoc` is also used to calculate association measures for all the variable pairs at different levels of a categorical variable. This helps in exploring the conditional associations and find out the differences between the groups of the conditioning variable. The function has a `by` argument which is used as the grouping variable and needs to be categorical.

```
iris_assoc_by <- calc_assoc(d = iris,
                            by = "Species")
iris_assoc_by
```

```
#> # A tibble: 24 x 5
#>    x            y            measure measure_type by
#>    <chr>        <chr>          <dbl> <chr>        <fct>
#>  1 Sepal.Width  Sepal.Length   0.743 pearson      setosa
#>  2 Petal.Length Sepal.Length   0.267 pearson      setosa
#>  3 Petal.Width  Sepal.Length   0.278 pearson      setosa
#>  4 Petal.Length Sepal.Width    0.178 pearson      setosa
#>  5 Petal.Width  Sepal.Width    0.233 pearson      setosa
#>  6 Petal.Width  Petal.Length   0.332 pearson      setosa
#>  7 Sepal.Width  Sepal.Length   0.526 pearson      versicolor
#>  8 Petal.Length Sepal.Length   0.754 pearson      versicolor
#>  9 Petal.Width  Sepal.Length   0.546 pearson      versicolor
#> 10 Petal.Length Sepal.Width    0.561 pearson      versicolor
#> # ... with 14 more rows
```

By default, the function `calc_assoc` calculates the association measures for all the variable pairs at different levels of the grouping variable and the pairwise association measures for the ungrouped data (`overall`) when used with the `by` argument. This behavior can be changed by setting `include.overall` argument to `FALSE`.

```
iris_assoc_by <- calc_assoc(d = iris,
                            by = "Species",
                            include.overall = FALSE)
iris_assoc_by
```

```
#> # A tibble: 18 x 5
#>    x            y            measure measure_type by
#>    <chr>        <chr>          <dbl> <chr>        <fct>
#>  1 Sepal.Width  Sepal.Length   0.743 pearson      setosa
#>  2 Petal.Length Sepal.Length   0.267 pearson      setosa
#>  3 Petal.Width  Sepal.Length   0.278 pearson      setosa
#>  4 Petal.Length Sepal.Width    0.178 pearson      setosa
#>  5 Petal.Width  Sepal.Width    0.233 pearson      setosa
#>  6 Petal.Width  Petal.Length   0.332 pearson      setosa
#>  7 Sepal.Width  Sepal.Length   0.526 pearson      versicolor
#>  8 Petal.Length Sepal.Length   0.754 pearson      versicolor
#>  9 Petal.Width  Sepal.Length   0.546 pearson      versicolor
#> 10 Petal.Length Sepal.Width    0.561 pearson      versicolor
#> 11 Petal.Width  Sepal.Width    0.664 pearson      versicolor
#> 12 Petal.Width  Petal.Length   0.787 pearson      versicolor
#> 13 Sepal.Width  Sepal.Length   0.457 pearson      virginica
#> 14 Petal.Length Sepal.Length   0.864 pearson      virginica
#> 15 Petal.Width  Sepal.Length   0.281 pearson      virginica
#> 16 Petal.Length Sepal.Width    0.401 pearson      virginica
#> 17 Petal.Width  Sepal.Width    0.538 pearson      virginica
#> 18 Petal.Width  Petal.Length   0.322 pearson      virginica
```

The tibble output in the conditional setting has a similar structure as `calc_assoc` used with no `by` argument. When used with the `by` argument, an additional `by` column representing the levels of the

Table 4: Variable description of a subset of the German election result dataset from 2002 and 2005.

Variable	Description
SPD.02	Proportion of votes for SPD in 2002
CDU.CSU.02	Proportion of votes for CDU/CSU in 2002
Gruene.02	Proportion of votes for Gruene in 2002
Pop.18.25	population between 18 and 25 years old 2002-12-31 (in percent)
Pop.25.35	population between 25 and 35 years old 2002-12-31 (in percent)
Pop.35.60	population between 35 and 60 years old 2002-12-31 (in percent)
Industry	industry employees subject to social insurance contribution (in percent)
CTT	commerce, transportation and telecommunication employees subject to social insurance contribution (in percent)
Unemployment.03	unemployment 2003-12-31 (in percent)

Table 5: Variable description of the CDC dataset

Variable	Description	VariableType
genhlth	General health, with categories excellent, very good, good, fair, and poor	ordinal
exerany	Respondent exercised in the past month with category 0 and 1	ordinal
hlthplan	Respondent has some form of health coverage	nominal
smoke100	Respondent has smoked at least 100 cigarettes in their entire life with category 0 and 1	ordinal
height	Respondent's height in inches	numerical
weight	Respondent's weight in pounds	numerical
wtdesire	Respondent's desired weight	numerical
age	Respondent's age in categories [18,25], (25,35], (35,60], (60,99]	ordinal
gender	Respondent's gender	nominal

categorical variable is added to the tibble output. The `x` and `y` variables in the output are repeated for every level of `by` variable. In order to have multiple `by` variables, the function `calc_assoc` is used multiple times with a different `by` variable each time and then the multiple outputs are binded row wise.

Section 5: corVis: Visualising Association

This section provides a detailed description of the novel visualisation techniques proposed in the package **corVis**. These methods display association and conditional association for every variable pair in a dataset in a single plot and show multiple bivariate measures of association simultaneously. The package includes functions such as `plot_assoc_matrix` and `plot_assoc_linear` to produce these displays in matrix and linear layout respectively. In addition, the package also provides a function `show_assoc` to display a scatterplot for a numeric variable pair input, a box plot for mixed variable pair input and bar plot for other variable pair input.

We use two datasets to provide illustrative examples. The first dataset is `de_elect` (German Election Data from 2002 and 2005) from **zenplots** package which includes numeric variables only. The German Election dataset provides the information on election results for two German elections held in 2002 and 2005. The dataset includes 299 constituencies and 68 variables providing information on these constituencies. For our analysis, we use a subset of variables from German election dataset which are described in Table 4. The poor economic performance of the country was one of the dominant issues during 2002 German election. To analyse the same, we use variables including vote proportion for three major parties in 2002 election, population percentages for different age groups, percentage of employees in different sectors subjected to social insurance contribution and percentage unemployment.

`cdc` (Behavioral survey data) is the second dataset from CDC (2000) which includes numeric, nominal and ordered variables. The dataset is a random sample of 20,000 people from Behavioral Risk Factor Surveillance System (BRFSS) survey, which is an annual telephonic survey of 350,000 people, in United States collected by the Centers for Disease Control and Prevention (CDC) conducted in 2000. The respondents of the survey are asked questions related to their diet, weekly physical activity and even their level of health coverage. The goal of the survey is to identify risk factors in adult population and report the health trends. The dataset used here has only 9 questions or variables as compared to the original dataset which includes more than 200 variables.Table 5 provides a brief description of the variables in `cdc` dataset along with the variable type. The variables `genhlth`, `exerany`, `smoke100` and `age` in the dataset were converted to ordinal factors as they had a natural ordering. `hlthplan` and `gender` were considered as nominal factors for the analysis. During the association analysis, we found some individuals with very high values for `height`, `weight` and `wtdesire`, and filtered out these cases for further analysis.

not expressed clearly

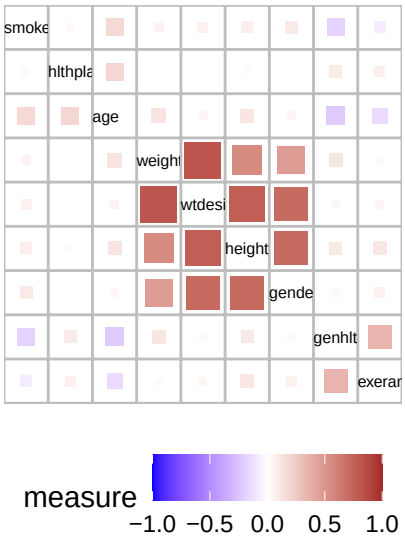

Figure 2: Association matrix display for cdc data showing Pearson's correlation for the numeric variable pairs, Goodman Kruskal's gamma measure for ordered variable pairs, canonical correlation for mixed variable pairs, nominal variable pairs and other variable pairs. The off diagonal cells show the measure value for a variable pair using a square glyph. The color of every square is mapped with the measure value for the variable pair and the area of the square is mapped by absolute measure value for the corresponding variable pair. The plot shows a strong association between desired weight and gender of the indvividuals. Also, there is an negative association between the general health and the age of the individuals suggesting the health of individuals deteriorating with age.

Association plots

For association analysis, we start with calculating the default association measures for the `cdc` data using `calc_assoc` and then this result is plotted using `plot_assoc_matrix` in a matrix layout in Figure 2.

```
assoc_cdc <- calc_assoc(d = cdcdata)
plot_assoc_matrix(lassoc = assoc_cdc)
```

The diagonal cells in Figure 2 represent the variables present in the data. Every off diagonal cell contains a glyph, square in this plot, which is filled with a divergent color scale representing the value of corresponding association measure for a variable pair. The `glyph` argument can be either `square` or `circle`. The area of the square is mapped to absolute value of the association measure which quickly highlights the associated pairs of variables. We also offer ordering of the variables in this display so that highly-associated variables are arranged closer to each other and the task of detecting patterns or relations becomes easier. The argument `var_order` is used for the variables in the matrix display. The function uses average linkage hierarchical clustering of the association matrix of the variables for ordering the variables, which clusters the highly associated variables together and arranges them nearby.

Figure 2 shows a high positive Pearson's correlation for (`wtdesire`, `height`) which suggests that taller individuals have higher desired weights. The plot also shows a negative Goodman Kruskal's gamma measure for the ordered variable pairs (`genhlth`, `age`) and (`genhlth`, `smoke100`) indicating that the health of individuals deteriorates as they age and with more smoking. The positive association for (`smoke100`, `age`) implies that older people tend to smoke more. Also, it is evident from the plot that individuals who exercise often are more healthier, shown by the variable pair (`exerany`, `genhlth`).

In order to explore these variable pairs in more detail, the function `show_assoc` is used to plot a scatterplot and a mosaic plot for numeric and ordinal pairs respectively in Figure 3.

```
show_assoc(d = cdcdata,
           x = "wtdesire",
           y = "height")

show_assoc(d = cdcdata,
           x = "age",
           y = "genhlth")
```

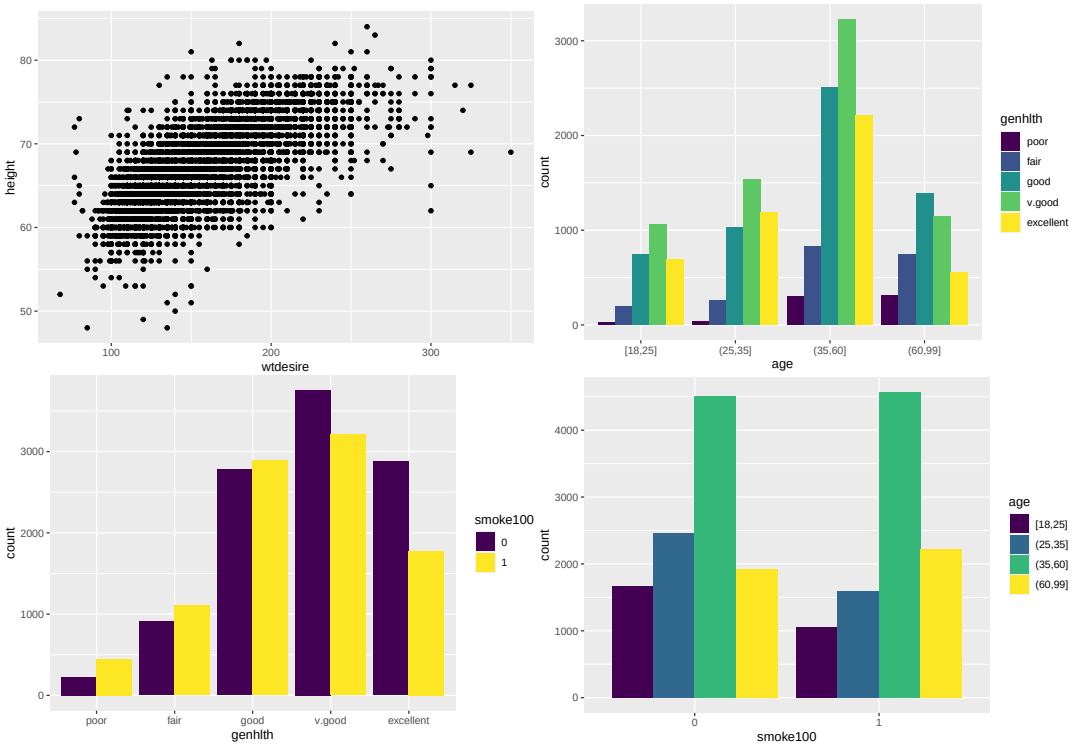

Figure 3: Scatterplot and barplot for numeric variable pair (wtdesire, height) and ordinal variable pairs (genhlth, age), (genhlth,smoke100) and (smoke100, age) showing association between these pair of variables.

```
show_assoc(d = cdcdata,
           x = "genhlth",
           y = "smoke100")

show_assoc(d = cdcdata,
           x = "smoke100",
           y = "age")
```

Multiple Association Measures Plot

The multiple association measures plot compares the multiple association measures for all the variable pairs in a dataset. This display is useful in detecting variable pairs with high difference among the measures which then can be explored further in more detail.

```
assoc_german_all <- calc_assoc_all(d = german_election,
                                   measures = c("ace",
                                                "pearson",
                                                "spearman",
                                                "kendall",
                                                "dcor",
                                                "spearman",
                                                "mic"))

plot_assoc_linear(assoc = assoc_german_all,
                  plot_type = "heatmap",
                  limits = c(0,1),
                  var_order = "max_diff")
```

Figure 4 shows a multiple association measures plot in linear layout for German election dataset. The plot compares the absolute values of association measures such as `ace`, `dcor`, `kendall`, `mic`, `pearson` and `spearman` for every variable pair in the dataset. Each cell of the plot corresponds to a variable pair and an association measure, and color intensity of each cell corresponds to the absolute value of association measure. The variable pairs in the plot are ordered by the maximum difference between the

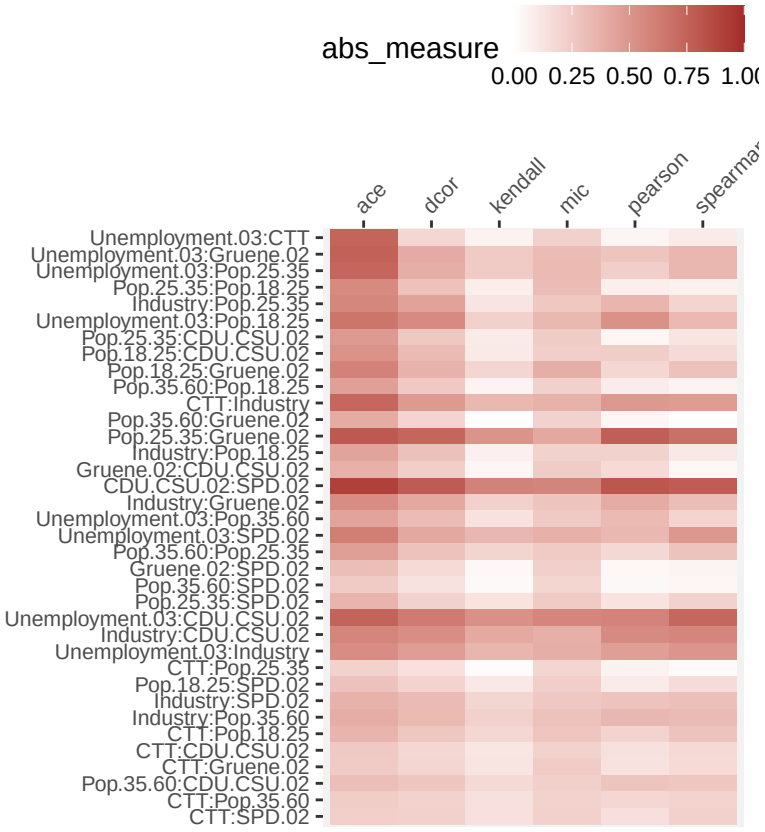

Figure 4: Multiple association measures plot in a linear layout for a subset of German election data. The plot has variable pairs on the Y-axis and association measures on the X-axis. The color intensity of each cell is proportional to the absolute value of association measure. The variable pairs on Y-axis are ordered by the maximum difference between the absolute value of association measures. The plot shows the highest difference for variable pair V.FDP.02 and V.Linke.02 which can be explored further to understand the underlying reasons for this difference.

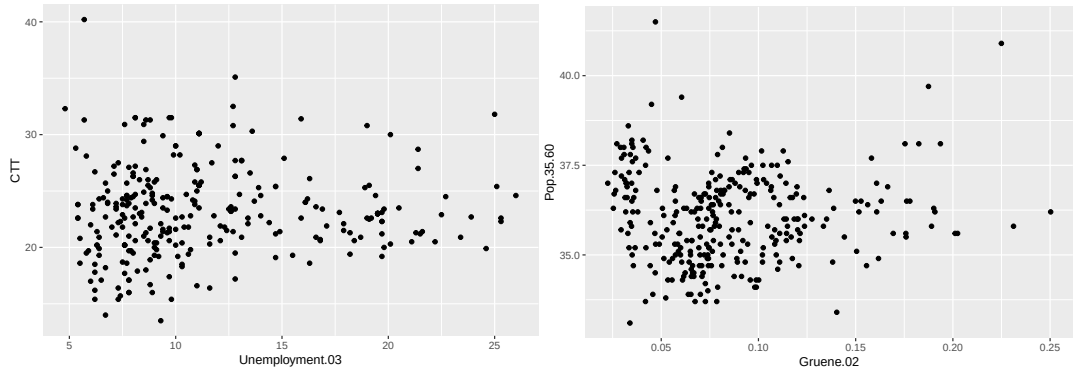

Figure 5: Scatterplot for variable pairs (from left to right) (Unemployment.03 and CTT) and ('Pop.35.60', 'Gruene.02') showing relationships between these pair of variables.

absolute value of these measures. The plot shows the variable pair (`Unemployment.03`,`CTT`) with highest difference between the association measures. The low value for Pearson's correlation and Kendall's correlation suggest no trend but measures such as ace, distance correlation, MIC and Spearman's correlation indicates that a relationship might exist among the variables. Another interesting variable pair evident from the plot is (`Pop.35.60`, `Gruene.02`) for which the three popular measures like Pearson's, Kendall's and Spearman's correlation coefficient are almost zero but measures such as ace, distance correlation and MIC suggest presence of a pattern.

```
show_assoc(d = german_election,
           x = "Unemployment.03",
           y = "CTT")

show_assoc(d = german_election,
           x = "Gruene.02",
           y = "Pop.35.60")
```

We use `show_assoc` to explore the relationship for the interesting variable pairs in Figure 5. It is evident from the plots that the variable pairs (`Unemployment.03`,`CTT`) and (`Pop.35.60`, `Gruene.02`) show a non-linear trend for which measures such as Pearson's correlation, Kendall's correlation and Spearman's correlation might not be suitable.

Conditional Association Plot

The conditional association plot is produced by splitting the data by a partitioning variable and calculating association for the variable pairs at each level of partitioning variable using `calc_assoc` function with conditioning variable as the `by` argument. The calculated association measures are then displayed using bars in a matrix plot. The height and color of the bars are coded with the value of association measure and the level of the partitioning variable respectively. These displays are efficient for discovering variable pair with high differences among the levels of partitioning variable in the data.

```
cond_assoc_cdc <- calc_assoc(d = cdcdata,
                             by = "genhlth")
plot_assoc_matrix(lassoc = cond_assoc_cdc)
```

Figure 6 shows a conditional association plot for the cdc data. Each cell corresponding to a variable pair shows two bars which correspond to the association measure (Pearson's correlation for numeric pairs, Goodman and Kruskal's gamma for ordinal pair, canonical correlation for nominal or mixed pairs) calculated at the levels of conditioning variable `genhlth`. The dotted line represents the overall association measure. The plot indicates that there is an evident difference in the Goodman and Kruskal's gamma for the variable pair (`smoke100`, `age`) for different levels of health, compared with each other and overall value. Also, the canonical correlation for variable pair (`weight`, `gender`) for individuals feeling poor or fair health is low compared to the overall value. We explore these variable pairs in more detail using `show_assoc`.

```
show_assoc(d = cdcdata,
           x = "smoke100",
           y = "age",
```

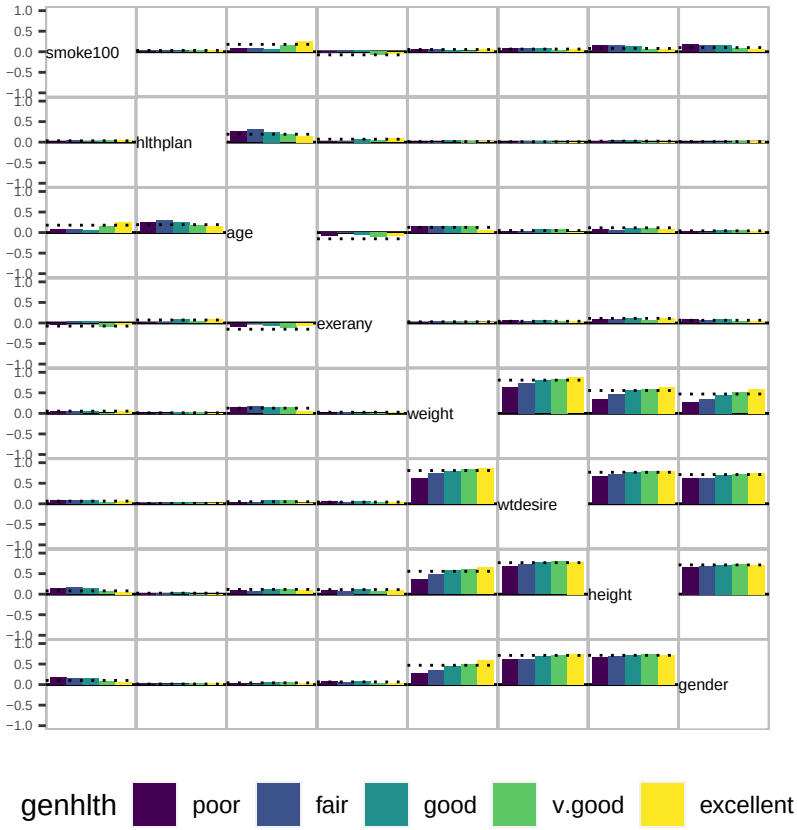

Figure 6: Conditional Association plot for cdc data showing Pearson's correlation for numeric pairs, Goodman and Kruskal's gamma for ordinal pair, canonical correlation for nominal or mixed pairs. The bars in each cell represent the value for asssociation measure colored by the conditioning variable genhlth The dotted line in each cell represents overall value of the association measure. The plot shows evident difference in measure value for pair (smoke100, age) and (weight, gender) for participants with different levels of health in the data.

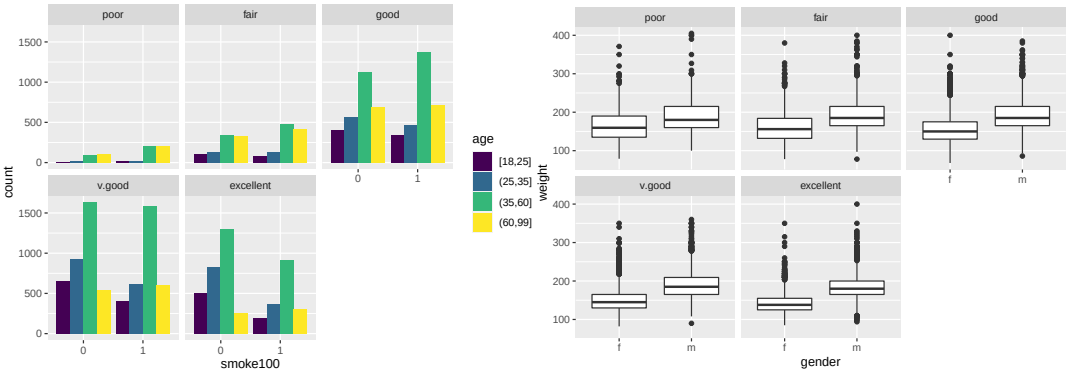

Figure 7: Barplot for variable pair (smoke100, age), and boxplot for variable pair (weight, gender) faceted by conditioning variable genhlth.

```
          by = "genhlth")

show_assoc(d = cdcdata,
          x = "weight",
          y = "gender",
          by = "genhlth")
```

Figure 7 shows a barplot for the variable pair (`smoke100`, `age`) and a boxplot for variable pair (`weight`, `gender`) faceted by the conditioning variable `genhlth`. The faceted barplot shows that individuals who are old with smoking habits suffer with poor health more (almost twice) compared to individuals who are old and don't smoke. Interstingly, the faceted boxplot shows that healthier females have low weight compared to the females who don't feel healthy. On the other hand, the weight of the males who feel either health or unhealthy have fairly similar weight.

We also use linear layouts for displaying conditional association in the package. The function `plot_assoc_linear` is used for displaying a linear layout of the conditional association for variable pairs in the dataset. The association measures are calculated for every variable pair at each level of partitioning variable using `calc_assoc` function with conditioning variable as the `by` argument.

The measures are then displayed using a dotplot (or a heatmap) where color of the dots (or each cell) is coded by the level of the partitioning variable and the variable pairs are ordered by absolute maximum value of association measure for each of the pair of variable. These displays are also efficient for discovering differences among the levels of partitioning variable in the data. In comparison to matrix layout, it is easier to omit less relevant pairs of variables in linear layouts by filtering the variables pairs having a higher value for association measures than a threshold.

Figure 8 shows a linear display for conditional association measures with the variable pairs having absoulte measure value greater than 0.1 along the Y-axis, the value of association measure along X-axis and color of the points representing the level of the grouping variable. The linear layout becomes more useful over the matrix layout for conditional association display when the number of variables and number of levels of grouping variable are high.

```
cond_assoc_cdc <- calc_assoc(d = cdcdata,
                             by = "genhlth")
cond_assoc_cdc <- dplyr::filter(cond_assoc_cdc, abs(measure) > 0.1)
plot_assoc_linear(assoc = cond_assoc_cdc,
                  plot_type = "dotplot")
```

Section 5: Discussion

We use multiple association measures in a single display for different variable pairs which serves as a comparison tool while exploring association in a dataset and assist in identifying unusual variable pairs. These multiple measures can be displayed in a scatterplot matrix similar to what Tukey and Tukey (1985) proposed. They suggested that scatterplot matrix of the scagnostics measures, which are measures summarizing a scatterplot, can be used to identify unusual scatterplots or variable pairs. Wilkinson et al. (2005) used this idea with their graph-theoretic scagnostic measures to highlight unusual scatterplots. Similarly, Kuhn et al. (2013) have used this idea in a predictive modeling context. They have produced a scatterplot matrix of the measures between the response and continuous

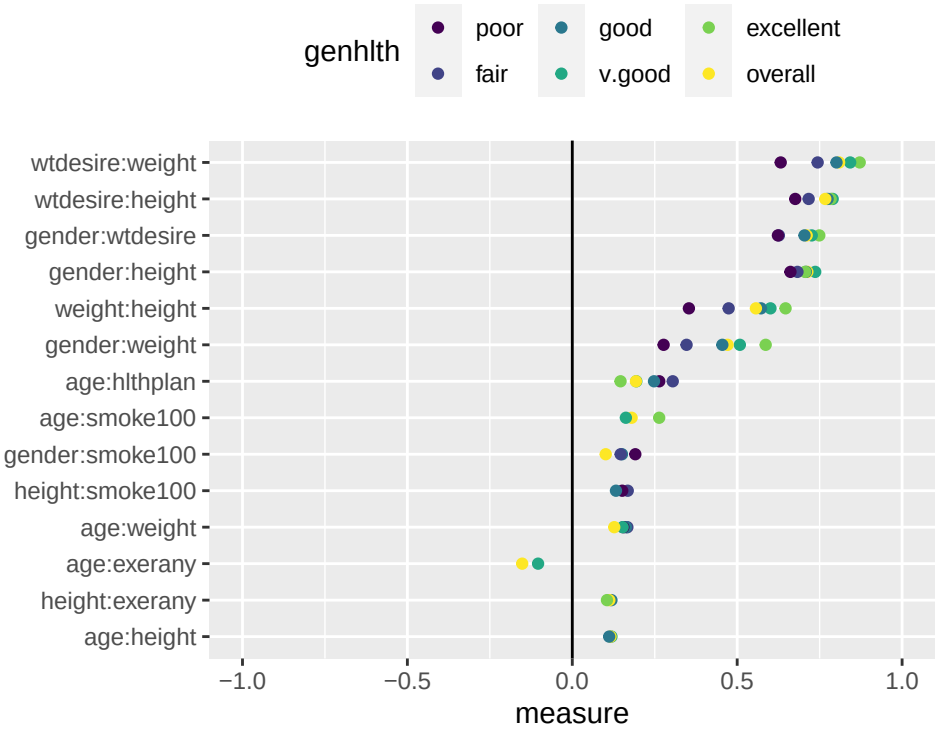

Figure 8: Conditional Association plot using linear layout.The display has variable pairs on the Y-axis and the value of association measures on the X-axis. The points corresponding to every variable pair represents the value of association measure for different levels of the conditioning variable and the overall value of association measure.

predictors such as Pearson's correlation coefficient, pseudo-R^2 from the locally weighted regression model, MIC and Spearman's rank correlation coefficient to explore the predictor importance during feature selection step. These displays show the importance of comparing multiple association measures at once for different variable pairs.

Bibliography

A. Agresti. *Analysis of ordinal categorical data*, volume 656. John Wiley & Sons, 2010. [p4]

A. Buja, A. M. Krieger, and E. I. George. A visualization tool for mining large correlation tables: The association navigator., 2016. [p2]

CDC. Behavioral risk factor surveillance system survey data, 2000. URL https://www.cdc.gov/brfss/. [p9]

M. Friendly. Corrgrams: Exploratory displays for correlation matrices. *The American Statistician*, 56(4): 316–324, 2002. [p1, 2]

S. Gerber. *scorr: s-CorrPlot: Visualizing Correlation*, 2022. URL http://mckennapsean.com/scorrplot/. R package version 1.0. [p2]

M. Hills. On looking at large correlation matrices. *Biometrika*, 56(2):249–253, 1969. [p2]

M. G. Kendall. The treatment of ties in ranking problems. *Biometrika*, 33(3):239–251, 1945. [p4]

M. Kuhn, K. Johnson, et al. *Applied predictive modeling*, volume 26. Springer, 2013. [p15]

M. Kuhn, S. Jackson, and J. Cimentada. *corrr: Correlations in R*, 2020. URL https://CRAN.R-project.org/package=corrr. R package version 0.4.3. [p2]

P. Morgen and P. Biecek. *corrgrapher: Explore Correlations Between Variables in a Machine Learning Model*, 2020. URL https://CRAN.R-project.org/package=corrgrapher. R package version 1.0.4. [p2]

D. J. Murdoch and E. Chow. A graphical display of large correlation matrices. *The American Statistician*, 50(2):178–180, 1996. [p2]

U. Olsson. Maximum likelihood estimation of the polychoric correlation coefficient. *Psychometrika*, 44 (4):443–460, 1979. [p4]

D. N. Reshef, Y. A. Reshef, H. K. Finucane, S. R. Grossman, G. McVean, P. J. Turnbaugh, E. S. Lander, M. Mitzenmacher, and P. C. Sabeti. Detecting novel associations in large data sets. *science*, 334(6062): 1518–1524, 2011. [p1, 3, 4]

A. Samba. *linkspotter: Bivariate Correlations Calculation and Visualization*, 2020. URL https://CRAN.R-project.org/package=linkspotter. R package version 1.3.0. [p2]

N. Simon and R. Tibshirani. Comment on "detecting novel associations in large data sets" by reshef et al, science dec 16, 2011, 2014. URL https://arxiv.org/abs/1401.7645. [p4]

T. Speed. A correlation for the 21st century. *Science*, 334(6062):1502–1503, 2011. [p4]

G. J. Székely, M. L. Rizzo, and N. K. Bakirov. Measuring and testing dependence by correlation of distances. *The annals of statistics*, 35(6):2769–2794, 2007. [p1, 3, 4]

H. Theil. On the estimation of relationships involving qualitative variables. *American Journal of Sociology*, 76(1):103–154, 1970. [p4]

E. R. Tufte. *The Visual Display of Quantitative Information*. Graphics Press, USA, 1986. ISBN 096139210X. [p1]

J. W. Tukey and P. A. Tukey. Computer graphics and exploratory data analysis: An introduction. In *Proceedings of the sixth annual conference and exposition: computer graphics*, volume 85, pages 773–785, 1985. [p1, 15]

T. Wei and V. Simko. *R package 'corrplot': Visualization of a Correlation Matrix*, 2021. URL https://github.com/taiyun/corrplot. (Version 0.92). [p2]

H. Wickham, M. Averick, J. Bryan, W. Chang, L. D. McGowan, R. François, G. Grolemund, A. Hayes, L. Henry, J. Hester, M. Kuhn, T. L. Pedersen, E. Miller, S. M. Bache, K. Müller, J. Ooms, D. Robinson, D. P. Seidel, V. Spinu, K. Takahashi, D. Vaughan, C. Wilke, K. Woo, and H. Yutani. Welcome to the tidyverse. *Journal of Open Source Software*, 4(43):1686, 2019. doi: 10.21105/joss.01686. [p2, 6]

L. Wilkinson, A. Anand, and R. Grossman. Graph-theoretic scagnostics. In *Information Visualization, IEEE Symposium on*, pages 21–21. IEEE Computer Society, 2005. [p15]

Amit Chinwan
Maynooth University
Hamilton Institute
Maynooth, Ireland
amit.chinwan.2019@mumail.ie

Catherine Hurley
Maynooth University
Department of Mathematics and Statistics
Maynooth, Ireland
catherine.hurley@mu.ie