ggcorrplot : Visualization of a Correlation Matrix using plotnine

version 0.0.2

Duvérier DJIFACK ZEBAZE

Contents

1 Introduction

1.1 What is ggcorrplot?

ggcorrplot is a library for visualization a correlation matrix. The **ggcorrplot** package can be used to **visualize easily** a **correlation matrix** using plotnine. It provides a solution for **reordering** the correlation matrix and displays the **significance level** on the correlogram. It includes also a function for computing a matrix of **correlation p-values**.

The only prerequisite for installing **ggcorrplot** is Python itself.

ggcorrplot can be installed with `pip`.

1.2 Installation

ggcorrplot can be installed from `pypi` as follow :

```
pip install ggcorrplot
```

1.3 Dependencies

ggcorrplot requires :

```
Python 3
numpy>=1.24.2
pandas>=2.0.0
plotnine>=0.10.1
scipy>=1.10.1
plydata>=0.4.3
```

1.4 Usage

Find out more https://github.com/enfantbenidedieu/ggcorrplot/blob/master/ggcorrplot.ipynb.

2 Code Reference

ggcorrplot provides multiple functions.

2.1 get_melt

Unpivot a DataFrame from wide to long format, optionally leaving identifiers set.

```
get_melt(x)
```

Parameters :

- x (DataFrame) : DataFrame to melt.

Return :

- Unpivoted DataFrame.

```
# Example
from ggcorrplot import *
from plotnine.data import mtcars
from plydata import *

# head
print(mtcars >> head(6))
```

```
##                 name   mpg  cyl   disp   hp  ...   qsec  vs  am  gear  carb
## 0          Mazda RX4  21.0    6  160.0  110  ...  16.46   0   1     4     4
## 1      Mazda RX4 Wag  21.0    6  160.0  110  ...  17.02   0   1     4     4
## 2         Datsun 710  22.8    4  108.0   93  ...  18.61   1   1     4     1
## 3     Hornet 4 Drive  21.4    6  258.0  110  ...  19.44   1   0     3     1
## 4  Hornet Sportabout  18.7    8  360.0  175  ...  17.02   0   0     3     2
## 5            Valiant  18.1    6  225.0  105  ...  20.22   1   0     3     1
##
## [6 rows x 12 columns]
```

Set the DataFrame index using columns `name`.

```
# Set index
mtcars = mtcars.set_index('name')
print(mtcars >> head(6))
```

```
##                     mpg  cyl   disp   hp  drat  ...   qsec  vs  am  gear  carb
## name                                            ...
## Mazda RX4          21.0    6  160.0  110  3.90  ...  16.46   0   1     4     4
## Mazda RX4 Wag      21.0    6  160.0  110  3.90  ...  17.02   0   1     4     4
## Datsun 710         22.8    4  108.0   93  3.85  ...  18.61   1   1     4     1
## Hornet 4 Drive     21.4    6  258.0  110  3.08  ...  19.44   1   0     3     1
## Hornet Sportabout  18.7    8  360.0  175  3.15  ...  17.02   0   0     3     2
## Valiant            18.1    6  225.0  105  2.76  ...  20.22   1   0     3     1
##
## [6 rows x 11 columns]
```

```
# Correlation Matrix
corr = mtcars.corr(method = "pearson").round(2)
print(corr)
```

```
##         mpg   cyl  disp    hp  drat    wt  qsec    vs    am  gear  carb
## mpg    1.00 -0.85 -0.85 -0.78  0.68 -0.87  0.42  0.66  0.60  0.48 -0.55
## cyl   -0.85  1.00  0.90  0.83 -0.70  0.78 -0.59 -0.81 -0.52 -0.49  0.53
## disp  -0.85  0.90  1.00  0.79 -0.71  0.89 -0.43 -0.71 -0.59 -0.56  0.39
## hp    -0.78  0.83  0.79  1.00 -0.45  0.66 -0.71 -0.72 -0.24 -0.13  0.75
## drat   0.68 -0.70 -0.71 -0.45  1.00 -0.71  0.09  0.44  0.71  0.70 -0.09
## wt    -0.87  0.78  0.89  0.66 -0.71  1.00 -0.17 -0.55 -0.69 -0.58  0.43
## qsec   0.42 -0.59 -0.43 -0.71  0.09 -0.17  1.00  0.74 -0.23 -0.21 -0.66
## vs     0.66 -0.81 -0.71 -0.72  0.44 -0.55  0.74  1.00  0.17  0.21 -0.57
## am     0.60 -0.52 -0.59 -0.24  0.71 -0.69 -0.23  0.17  1.00  0.79  0.06
## gear   0.48 -0.49 -0.56 -0.13  0.70 -0.58 -0.21  0.21  0.79  1.00  0.27
## carb  -0.55  0.53  0.39  0.75 -0.09  0.43 -0.66 -0.57  0.06  0.27  1.00
```

```
# Unpivoted DataFrame
print(get_melt(corr))
```

```
##      Var1  Var2  value
## 0     mpg   mpg   1.00
## 1     mpg   cyl  -0.85
## 2     mpg  disp  -0.85
## 3     mpg    hp  -0.78
## 4     mpg  drat   0.68
## ..    ...   ...    ...
## 116  carb  qsec  -0.66
## 117  carb    vs  -0.57
## 118  carb    am   0.06
## 119  carb  gear   0.27
## 120  carb  carb   1.00
##
## [121 rows x 3 columns]
```

2.2 match_arg

Argument verification using partial matching.

```
match_arg(x)
```

Parameters :

- `x` (str) : string argument.
- `arg` (list) : a list of candidate values.

Return :

- The unabbreviated version of the exact or unique partial match if there is one.

```
# match arguments
lst = ["gaussian", "epanechnikov", "rectangular", "triangular"]
print(match_arg("gaussian", lst))
```

```
## gaussian
```

2.3 get_upper_tri

Get upper triangle of the correlation matrix.

```
get_upper_tri(cormat,show_diag = False)
```

Parameters :

- `cormat` (DataFrame) : Correlation Matrix.
- `show_diag` (bool) : boolean. If `True`, displays the correlation coefficients.

Return :

- Upper triangle of a correlation matrix.

```
# show_diag = False
print(get_upper_tri(corr,show_diag = False))
```

```
##        mpg   cyl  disp    hp  drat    wt  qsec    vs    am  gear  carb
## mpg    NaN -0.85 -0.85 -0.78  0.68 -0.87  0.42  0.66  0.60  0.48 -0.55
## cyl    NaN   NaN  0.90  0.83 -0.70  0.78 -0.59 -0.81 -0.52 -0.49  0.53
## disp   NaN   NaN   NaN  0.79 -0.71  0.89 -0.43 -0.71 -0.59 -0.56  0.39
## hp     NaN   NaN   NaN   NaN -0.45  0.66 -0.71 -0.72 -0.24 -0.13  0.75
## drat   NaN   NaN   NaN   NaN   NaN -0.71  0.09  0.44  0.71  0.70 -0.09
## wt     NaN   NaN   NaN   NaN   NaN   NaN -0.17 -0.55 -0.69 -0.58  0.43
## qsec   NaN   NaN   NaN   NaN   NaN   NaN   NaN  0.74 -0.23 -0.21 -0.66
## vs     NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN  0.17  0.21 -0.57
```

```
## am    NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN  0.79  0.06
## gear  NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN  0.27
## carb  NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN
```

```
# show_diag = True
print(get_upper_tri(corr,show_diag = True))
```

```
##        mpg   cyl  disp    hp  drat    wt  qsec    vs    am  gear  carb
## mpg    1.0 -0.85 -0.85 -0.78  0.68 -0.87  0.42  0.66  0.60  0.48 -0.55
## cyl    NaN  1.00  0.90  0.83 -0.70  0.78 -0.59 -0.81 -0.52 -0.49  0.53
## disp   NaN   NaN  1.00  0.79 -0.71  0.89 -0.43 -0.71 -0.59 -0.56  0.39
## hp     NaN   NaN   NaN  1.00 -0.45  0.66 -0.71 -0.72 -0.24 -0.13  0.75
## drat   NaN   NaN   NaN   NaN  1.00 -0.71  0.09  0.44  0.71  0.70 -0.09
## wt     NaN   NaN   NaN   NaN   NaN  1.00 -0.17 -0.55 -0.69 -0.58  0.43
## qsec   NaN   NaN   NaN   NaN   NaN   NaN  1.00  0.74 -0.23 -0.21 -0.66
## vs     NaN   NaN   NaN   NaN   NaN   NaN   NaN  1.00  0.17  0.21 -0.57
## am     NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN  1.00  0.79  0.06
## gear   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN  1.00  0.27
## carb   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN  1.00
```

2.4 get_lower_tri

Get lower triangle of the correlation matrix.

```
get_lower_tri(cormat,show_diag=False)
```

Parameters :

- `cormat` (DataFrame) : Correlation Matrix.
- `show_diag` (bool) : boolean. If `True`, displays the correlation coefficients.

Return :

- Lower triangle of a correlation matrix.

```
# show_diag = False
get_lower_tri(corr,show_diag=False)
```

```
##          mpg   cyl  disp    hp  drat    wt  qsec    vs    am  gear  carb
## mpg      NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN
## cyl    -0.85   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN
## disp   -0.85  0.90   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN
## hp     -0.78  0.83  0.79   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN
## drat    0.68 -0.70 -0.71 -0.45   NaN   NaN   NaN   NaN   NaN   NaN   NaN
## wt     -0.87  0.78  0.89  0.66 -0.71   NaN   NaN   NaN   NaN   NaN   NaN
## qsec    0.42 -0.59 -0.43 -0.71  0.09 -0.17   NaN   NaN   NaN   NaN   NaN
## vs      0.66 -0.81 -0.71 -0.72  0.44 -0.55  0.74   NaN   NaN   NaN   NaN
## am      0.60 -0.52 -0.59 -0.24  0.71 -0.69 -0.23  0.17   NaN   NaN   NaN
## gear    0.48 -0.49 -0.56 -0.13  0.70 -0.58 -0.21  0.21  0.79   NaN   NaN
## carb   -0.55  0.53  0.39  0.75 -0.09  0.43 -0.66 -0.57  0.06  0.27   NaN
```

```
# show_diag = True
get_lower_tri(corr,show_diag=True)
```

```
##        mpg   cyl  disp    hp  drat    wt  qsec    vs    am  gear  carb
## mpg   1.00   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN
## cyl  -0.85  1.00   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN
## disp -0.85  0.90  1.00   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN
## hp   -0.78  0.83  0.79  1.00   NaN   NaN   NaN   NaN   NaN   NaN   NaN
## drat  0.68 -0.70 -0.71 -0.45  1.00   NaN   NaN   NaN   NaN   NaN   NaN
## wt   -0.87  0.78  0.89  0.66 -0.71  1.00   NaN   NaN   NaN   NaN   NaN
## qsec  0.42 -0.59 -0.43 -0.71  0.09 -0.17  1.00   NaN   NaN   NaN   NaN
## vs    0.66 -0.81 -0.71 -0.72  0.44 -0.55  0.74  1.00   NaN   NaN   NaN
## am    0.60 -0.52 -0.59 -0.24  0.71 -0.69 -0.23  0.17  1.00   NaN   NaN
## gear  0.48 -0.49 -0.56 -0.13  0.70 -0.58 -0.21  0.21  0.79  1.00   NaN
## carb -0.55  0.53  0.39  0.75 -0.09  0.43 -0.66 -0.57  0.06  0.27   1.0
```

2.5 cor_pmat

Compute a correlation matrix p-values.

```
cor_pmat(x,**kwargs)
```

Parameters :

- `x` (DataFrame) : DataFrame containing multiple variables and observations. Each column represents a variable, and each row a single observation of all those variables.
- `**kwargs` : other arguments to be passed to the function pearsonr.

Return :

- DataFrame containing the p-values of correlations.

```
# Computing correlation matrix with p-values
cor_pmat(mtcars)
```

```
##                mpg           cyl  ...          gear          carb
## mpg   0.000000e+00  6.112687e-10  ...  5.400948e-03  1.084446e-03
## cyl   6.112687e-10  0.000000e+00  ...  4.173297e-03  1.942340e-03
## disp  9.380327e-10  1.802838e-12  ...  9.635921e-04  2.526789e-02
## hp    1.787835e-07  3.477861e-09  ...  4.930119e-01  7.827810e-07
## drat  1.776240e-05  8.244636e-06  ...  8.360110e-06  6.211834e-01
## wt    1.293959e-10  1.217567e-07  ...  4.586601e-04  1.463861e-02
## qsec  1.708199e-02  3.660533e-04  ...  2.425344e-01  4.536949e-05
## vs    3.415937e-05  1.843018e-08  ...  2.579439e-01  6.670496e-04
## am    2.850207e-04  2.151207e-03  ...  5.834043e-08  7.544526e-01
## gear  5.400948e-03  4.173297e-03  ...  0.000000e+00  1.290291e-01
## carb  1.084446e-03  1.942340e-03  ...  1.290291e-01  0.000000e+00
##
## [11 rows x 11 columns]
```

2.6 remove_diag

Fill the main diagonal of the correlation matrix with NA.

```
remove_diag(cormat)
```

Parameters :

- `cormat` (DataFrame) : Correlation Matrix.

Return :

- This function modifies the input array in-place.

```
# Remove diagonal
print(remove_diag(corr))
```

```
##        mpg   cyl  disp    hp  drat    wt  qsec    vs    am  gear  carb
## mpg    NaN -0.85 -0.85 -0.78  0.68 -0.87  0.42  0.66  0.60  0.48 -0.55
## cyl  -0.85   NaN  0.90  0.83 -0.70  0.78 -0.59 -0.81 -0.52 -0.49  0.53
## disp -0.85  0.90   NaN  0.79 -0.71  0.89 -0.43 -0.71 -0.59 -0.56  0.39
## hp   -0.78  0.83  0.79   NaN -0.45  0.66 -0.71 -0.72 -0.24 -0.13  0.75
## drat  0.68 -0.70 -0.71 -0.45   NaN -0.71  0.09  0.44  0.71  0.70 -0.09
## wt   -0.87  0.78  0.89  0.66 -0.71   NaN -0.17 -0.55 -0.69 -0.58  0.43
## qsec  0.42 -0.59 -0.43 -0.71  0.09 -0.17   NaN  0.74 -0.23 -0.21 -0.66
## vs    0.66 -0.81 -0.71 -0.72  0.44 -0.55  0.74   NaN  0.17  0.21 -0.57
## am    0.60 -0.52 -0.59 -0.24  0.71 -0.69 -0.23  0.17   NaN  0.79  0.06
## gear  0.48 -0.49 -0.56 -0.13  0.70 -0.58 -0.21  0.21  0.79   NaN  0.27
## carb -0.55  0.53  0.39  0.75 -0.09  0.43 -0.66 -0.57  0.06  0.27   NaN
```

2.7 ggcorrplot

A graphical display of a Correlation Matrix using plotnine.

```
ggcorrplot(x,method = "square",type = "full",ggtheme = plotnine.theme_minimal(),
           title = None,show_legend = True,legend_title = "Corr",show_diag = None,
           colors = ["blue","white","red"],outline_color = "gray",hc_order = False,
           hc_method = "complete",lab = False,lab_col = "black",lab_size = 11,
           p_mat = None,sig_level=0.05,insig = "pch",pch = 4,pch_col = "black",
           pch_cex = 5,tl_cex = 12,tl_col = "black",tl_srt = 45,digits = 2)
```

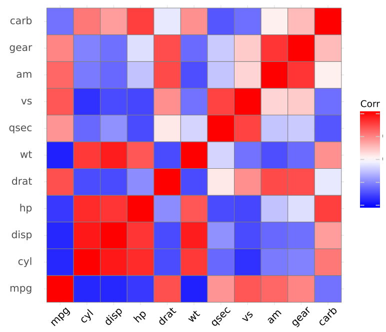
Parameters :

- `x` (DataFrame) : DataFrame containing multiple variables and observations. Each column represents a variable, and each row a single observation of all those variables.
- `method` (str) : the visualization method of correlation matrix to be used. Allowed values are `square` (default), `circle`.
- `type` (str) : `full` (default), `lower` or `upper` display.

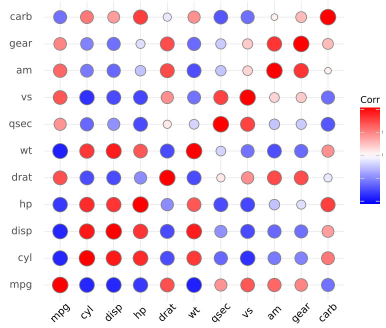
- `ggtheme` (theme) : plotnine function. Default value is `theme_minimal`. Allowed values are the official plotnine themes including `theme_gray`, `theme_bw`, `theme_minimal`, `theme_classic`, `theme_void`.
- `title` (str) : title of the graph
- `show_legend` (bool) : if `True` the legend is displayed.
- `legend_title` (str) : legend title. lower triangular, upper triangular or full matrix.
- `show_diag` (None|bool) : Whether display the correlation coefficients on the principal diagonal. If `None`, the default is to show diagonal correlation for `type=full` and to remove it when the type is one of `upper` or `lower`.
- `colors` (list) : a list of 3 colors for low, mid and high correlation values.
- `outline_color` (str) : the outline color of squared or circle. Default value is `gray`.
- `hc_order` (bool) : if `True`, correlation matrix will be hc_ordered using linkage function.
- `hc_method` (str) : the linkage method to be used in linkage function.
- `lab` (bool) : if `True`, add correlation coefficient on the plot.
- `lab_col` (str) : color to be used for the correlation coefficient labels, used when `lab=True`.
- `lab_size` (int) : size to be used for correlation coefficient labels, used when `lab=True`.
- `p_mat` (DataFrame) : DataFrame of p-value. If `None`, arguments `sig_level`, `insig`, `pch`, `pch_col`, `pch_cex` is invalid.
- `sig_level` (float) :significant level, if the p-value in p_mat is bigger that sig_level, then the corresponding correlation coefficient is regarded as insignificant.
- `insig` (str) : specialized insignificant correlation coefficients, `pch` (default), `blank`. If `blank`, wipe away the corresponding glyphs; if `pch`, add string (see `pch` for details) on corresponding glyphs.
- `pch` (int) : add string on the glyphs of insignificant correlation coefficients (only valid when insig is `pch`). Default value is 4.
- `pch_col` (str) : the color of pch (only valid when insig is `pch`).
- `pch_cex` (int) : the cex (size) of pch (only valid when insig is `pch`).
- `tl_cex` (int) : the size of text label (variable names).
- `tl_col` (str) : the color of text label (variable names).
- `tl_srt` (int) : the integer rotation of text label (variable names).
- `digits` (int) : Decides the number of decimal digits to be displayed (Default : `2`).

Visualizing the correlation matrix using different methods

```
# Visualizing the correlation matrix using "square" (default) method
p = ggcorrplot(mtcars,method="square")
print(p)
```

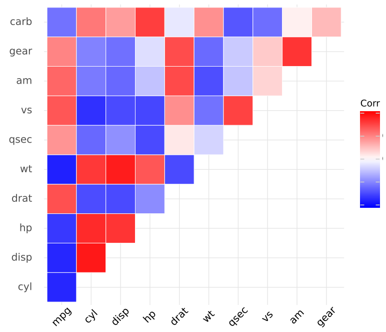


```
# Visualizing the correlation matrix using "circle" method
p = ggcorrplot(mtcars,method="circle")
print(p)
```

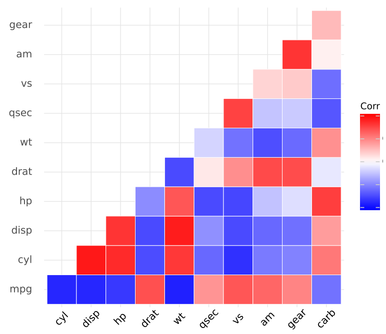


Visualizing correlation matrix using different layouts

```
# Visualizing upper triangle layouts
p = ggcorrplot(mtcars,type ="upper",outline_color ="white")
print(p)
```

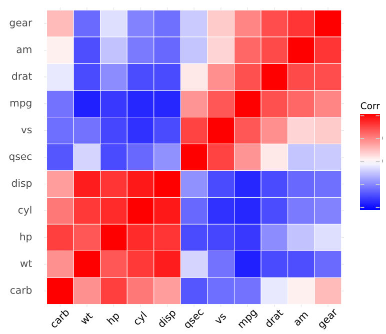



```
# Visualizing lower triangle layouts
p = ggcorrplot(mtcars,type ="lower",outline_color ="white")
print(p)
```



Reordering of the correlation matrix

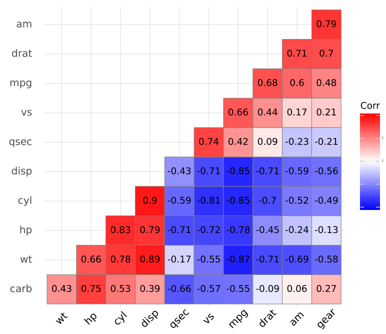
```
# Visualizing and reordering correlation matrix
p = ggcorrplot(mtcars, hc_order =True,outline_color ="white")
print(p)
```



Introducing correlation coefficient

We will now visualize our correlation matrix by adding the correlation coefficient using the **ggcorrplot** function and providing correlation matrix, `hc_order`, `type`, and `lower` variables as arguments.

```
# Adding the correlation coefficient
p = ggcorrplot(mtcars, hc_order =True, type ="lower",lab =True)
print(p)
```

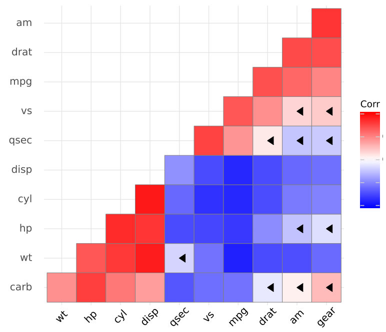


Adding significance level

Basically, the significance level is denoted by alpha. We compare the significance level to p-values to check whether the correlation between variables is significant or not. If p-value is less than equal to alpha, then the correlation is significant else, non-significant.

We will visualize our correlation matrix by adding significance level not taking any significant coefficient. We will do this using the **ggcorrplot** function and taking arguments as our correlation matrix, `hc_order`, `type`, and our correlation matrix with p-values.

```
# Computing correlation matrix with p-values
corrp_mat = cor_pmat(mtcars)
# Adding correlation significance level
p = ggcorrplot(mtcars,hc_order=True, type ="lower",p_mat = corrp_mat)
print(p)
```

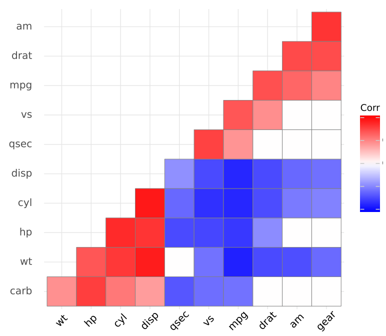


Leaving blank on no significance level

We will now visualize our correlation matrix by leaving a blank where there is no significance level. In the previous example, we added a significance level to our correlation matrix. Here, we will remove those parts of the correlation matrix where we did not find any significance level.

We will do this using the **ggcorrplot** function and take arguments like our correlation matrix, correlation matrix with p-values, `hc_order`, `type` and `insig`.

```
# Leaving blank on no significance level
p = ggcorrplot(mtcars,hc_order=True,type ="lower",p_mat=corrp_mat,insig="blank")
print(p)
```



References

Harris, Charles R., K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, et al. 2020. "Array Programming with NumPy." *Nature* 585 (7825): 357–62. https://doi.org/10.1038/s41586-020-2649-2.

team, The pandas development. 2020. *Pandas-Dev/Pandas: Pandas* (version latest). Zenodo. https://doi.org/10.5281/zenodo.3509134.

Virtanen, Pauli, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, et al. 2020. "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python." *Nature Methods* 17: 261–72. https://doi.org/10.1038/s41592-019-0686-2.