vivainsights R library: CHEAT SHEET (1/2)

Full documentation site:
https://microsoft.github.io/vivainsights/

Basics

vivainsights is an R library that offers a set of tools and functions for analysing and visualising data from **Microsoft Viva Insights**

You can install or update the package with the following command from CRAN:

```
install.packages("vivainsights")
```

The latest development version and documentation can be found on our GitHub repository: https://github.com/microsoft/vivainsights/

?function_name

To load the documentation in R, prefix the function name with a question mark

Tip!

Load **{tidyverse}** as the companion package to **Viva Insights** for seamless data wrangling

Example set up

```
library(vivainsights)
library(tidyverse)
pq_df <- import_query("path/…csv")

# Collaboration hours - summary
collab_sum(pq_df))
```

Data import / export

Use our handy functions below which are optimized for best practice for getting data *in* and *out* of R

- **import_query()**
  Import CSV queries faster and pre-formatted for **vivainsights** functions (instead of read.csv())
- **export()**
  Copy a data frame to clipboard, or write as a CSV, or a ggplot object as PNG or SVG

Data validation

Validate and understand your data prior to starting a piece of analysis

validation_report()
Generate a report to validate person query data, with options to supply an additional *meeting query*

hrvar_count()
Count number of employees in HR attribute

extract_hr()
Extract HR attributes in a query

check_query()
Print diagnostic data about the query to the R console

identify_holidayweeks()
Identify likely holiday weeks (for the entire pop) where collaboration hours lie far outside the mean

identify_nkw()
Identify likely non-knowledge workers where average person collaboration hours lie far outside the mean

identify_inactiveweeks()
Identify likely person-weeks where collaboration hours lie far outside the mean relative to the population average

identify_tenure()
Calculate tenure based on a supplied hire date

identify_outlier()
Takes in a selected metric and uses z-score (number of standard deviations) to identify outliers across time

identify_privacythreshold()
This function scans a standard query output for groups with of employees under the privacy threshold

hr_trend()
Returns a line chart showing the change in employee count over time

Inbuilt datasets

Explore **vivainsights** by using inbuilt demo datasets

pq_data
Person query

mt_data
Meeting query

p2p_data
Person-to-Person query

p2p_data_sim()
Person to Person query / edge list based on the graph (Watts-Strogatz small-world network model)

Exploratory analysis

Explore the data and surface initial hypotheses

keymetrics_scan(), keymetrics_scan_asis()
Returns a heatmapped table by default, with options to return a table; Return a heatmapped table directly from the aggregated / summarized data

create_rank()
Returns a plot by default, with an option to return a table with all groups (across multiple HR attributes) ranked by the specified metric

Distribution

Understand the distribution of a metric

create_boxplot()
Analyzes a selected metric and returns a box plot by default

create_density()
Returns a faceted density plot by default

create_hist()
Returns a faceted histogram by default

Basic analysis

Combine prefix with plot type to create a specific analysis on a Viva Insights metric
Available prefixes: **collab, email, meeting, afterhours, one2one, workloads**

***_summary()**
hrvar, mingroup, return
Returns a bar plot showing average weekly email hours by default

***_dist()**
hrvar, mingroup, return
Returns a stacked bar plot by default

***_fizz()**
hrvar, mingroup, return
Returns a 'fizzy scatter plot by default

***_line()**
hrvar, mingroup, return
Returns a line chart for email hours by default

***_trend()**
hrvar, mingroup, return
By default, returns a week-by-week heatmap, highlighting the time with most activity

***_rank()**
hrvar, mingroup
Returns a plot by default, with an option to return a table with a all of groups (across multiple HR attributes) ranked by hours of digital collaboration

vivainsights R library: CHEAT SHEET (2/2)

Full documentation site:
https://microsoft.github.io/vivainsights/

Flexible analysis

Flexible analysis functions are versatile, allowing you to pass any metric as a string parameter, e.g., **metric = 'Email_hours'**

create_bar(), create_bar_asis()
metric, hrvar, mingroup, return
Returns a bar plot showing the average of a selected metric by default. This function creates a bar chart directly from the aggregated / summarized data

create_fizz()
metric, hrvar, mingroup, return
Analyzes a selected metric and returns a 'fizzy' scatter plot by default

create_scatter()
metric, hrvar, mingroup, return
Returns a scatter plot of two selected metrics, using color to map an HR attribute

create_bubble()
metric, hrvar, mingroup, return
Returns a bubble plot of two selected metrics, using size to map the number of employees

create_dist()
metric, hrvar, mingroup, return
Returns a stacked bar plot by default

create_inc()
metric, hrvar, mingroup, return
Returns a heatmap for the generated incidence analysis

create_sankey()
data, var1, var2, count
Create a 'networkD3' style sankey chart based on a long count table with two variables

create_stacked()
metric, hrvar, mingroup, return
Returns a stacked bar plot by default

Flexible analysis - over time

Flexible analysis functions for understanding changes over time

create_line(), create_line_asis()
hrvar, mingroup
By default, returns a line chart for the defined metric

create_period_scatter()
metric_x, metric_y, hrvar, mingroup, return
Returns a faceted scatter plot by default

create_trend()
metric, hrvar, mingroup, return
By default, returns a week-by-week heatmap bar plot, highlighting the points in time with most activity

create_tracking()
metric, percent
Create a line chart that visualizes a set of metric over time for the selected population

Network analysis

Analyze edge list datasets (e.g., Person-to-person, Group-to-group) from Viva Insights

network_g2g()
Pass a data frame containing a group-to-group query and return a network plot

network_p2p()
Pass a data frame containing a person-to-person query and return a network visualization

network_summary()
Pass an igraph object to the function and obtain centrality statistics for each node in the object as a data frame

Other analysis

create_lorenz()
This function computes the Gini coefficient and plots the Lorenz curve based on a selected metric from a Person Query data frame

create_IV()
Specify an outcome variable and return IV outputs

IV_report()
The function generates an interactive HTML report using Standard Person Query data as an input. Report based on running the Information Value (IV) algorithm

maxmin()
This function allows you to scale vectors or an entire data frame using the max-min scaling method, always returning a numeric vector

Helper functions

anonymise(), anonymize()
Anonymize categorical variables such as HR variables by replacing values with dummy team names such as 'Team A'. The behavior is to make 1 to 1 replacements by default

totals_bind()
The purpose of this is to enable to creation of summary tables with a calculated "Total" row

totals_col()
Create a 'Total' column of character type comprising exactly of one unique value

tstamp()
This function generates a time stamp of the format `'yymmdd_hhmmss'`. This is a support function and is not intended for direct use

us_to_space()
Convenience function to convert underscores to space

wrap()
This function adds a character at the start and end of a character string, where the default behavior is to add a double quote

wrap_text()
Wrap text in visualizations according to a preset character threshold