

# Package ‘ggsurvey’

January 21, 2022

**Type** Package

**Title** ggplot2 for Survey Data

**Version** 0.1.0

**Author** Brittany Alexander

**Maintainer** Brittany Alexander <brittanyalexander1337@gmail.com>

**Description** This package provides code to make plots of weighted survey data

**License** GPL (>= 2)

**Encoding** UTF-8

**Imports** ggplot2, stats, dplyr, survey, hexbin

**LazyData** true

**RoxygenNote** 7.1.2

## R topics documented:

ggbarcrosstabs . . . . .	2
ggbarcrosstabs3d . . . . .	2
ggbarcrosstabs3d_svy . . . . .	3
ggbarcrosstabs_svy . . . . .	3
ggbarweight . . . . .	4
ggbarweight_svy . . . . .	4
ggboxweight . . . . .	5
ggboxweight2d . . . . .	6
ggboxweight2d_svy . . . . .	6
ggboxweight3d . . . . .	7
ggboxweight3d_svy . . . . .	7
ggboxweight_svy . . . . .	8
gghexweight . . . . .	8
gghexweight2d . . . . .	9
gghexweight2d_svy . . . . .	9
gghexweight3d . . . . .	10
gghexweight3d_svy . . . . .	10
gghexweight_svy . . . . .	11
gghistweight . . . . .	11
gghistweight2d . . . . .	12
gghistweight2d_svy . . . . .	12
gghistweight3d . . . . .	13

gghistweight3d\_svy . . . . . 13

gghistweight\_svy . . . . . 14

hello . . . . . 14

**Index** 15

---

ggbarcrosstabs	<i>Crosstabs of Two Variables</i>
----------------	-----------------------------------

---

**Description**

Crosstabs of Two Variables

**Usage**

ggbarcrosstabs(df, x, y, weights)

**Arguments**

- df data frame of survey
- x variable to bar chart
- y faceting variable
- weights survey weights that sum to sample size

**Value**

ggplot object that can be customized using ggplot2 functions including adding titles

**Examples**

```
library(survey)
data(api)
ggbarcrosstabs(apistrat, stype, yr.rnd, pw)
data(nhanes)
ggbarcrosstabs(nhanes, race, agecat, WTMEC2YR)
```

---

ggbarcrosstabs3d	<i>Crosstabs of Three Variables</i>
------------------	-------------------------------------

---

**Description**

Crosstabs of Three Variables

**Usage**

ggbarcrosstabs3d(df, x, y, z, weights)

**Arguments**

df	data frame
x	bar chart variable
y	crosstab variable 1 (horizontal facets)
z	crosstab variable 2 (vertical facets)
weights	survey weights that sum to sample size

**Examples**

```
library(survey)
data(api)
ggbarcrosstabs3d(apistrat, stype, yr.rnd, awards, pw)
data(nhanes)
ggbarcrosstabs3d(nhanes, race, agecat, RIAGENDR, WTMEC2YR)
```

---

ggbarcrosstabs3d\_svy    *Crosstabs of Three Variables Using svy.design object*

---

**Description**

Crosstabs of Three Variables Using svy.design object

**Usage**

```
ggbarcrosstabs3d_svy(surveyobj, x, y, z)
```

**Arguments**

surveyobj	svy.design obj
x	bar chart variable
y	crosstab variable 1 (horizontal facets)
z	crosstab variable 2 (vertical facets)

**Examples**

```
library(survey)
data(api)
dstrat<-svydesign(id=~1,strata=~stype, weights=~pw, data=apistrat, fpc=~fpc)
ggbarcrosstabs3d_svy(dstrat, stype, yr.rnd, awards)
data(nhanes)
design <- svydesign(id=~SDMVPSU, strata=~SDMVSTRA, weights=~WTMEC2YR, nest=TRUE,data=nhanes)+ylab("Proportion")
ggbarcrosstabs3d_svy(design, race, agecat, RIAGENDR)
```

---

ggbarcrosstabs_svy	<i>Crosstabs for svy.design objects</i>
--------------------	---

---

### Description

Crosstabs for svy.design objects

### Usage

```
ggbarcrosstabs_svy(surveyobj, x, y)
```

### Arguments

x	variable for bar chart
y	faceting variable (comparison factor)
svyobj	svy.design

### Examples

```
library(survey)
data(api)
dstrat<-svydesign(id=~1,strata=~stype, weights=~pw, data=apistrat, fpc=~fpc)
ggbarcrosstabs_svy(dstrat, stype, yr.rnd)+ylab("Proportion")
data(nhanes)
design <- svydesign(id=~SDMVPSU, strata=~SDMVSTRA, weights=~WTMEC2YR, nest=TRUE,data=nhanes)+ylab("Proportion")
ggbarcrosstabs_svy(design, race, agecat)
```

---

ggbarweight	<i>Weighted Univariate Bar Charts</i>
-------------	---------------------------------------

---

### Description

Weighted Univariate Bar Charts

### Usage

```
ggbarweight(df, x, weights)
```

### Arguments

df	data frame of survey
x	name of question of interest
weights	survey weights that sums to sample size

### Value

ggplot object that can be customized using ggplot2 functions including adding titles

**Examples**

```
library(survey)
#Example with data frame
data(api)
ggbarweight(apistrat, stype, pw)+ggtitle("Proportion of School Type")+ylab("Proportion")
data(nhanes)
ggbarweight(nhanes, race, WTMEC2YR)+ylab("Proportion")
```

ggbarweight\_svy

*Bar Chart from svydesign objects***Description**

Bar Chart from svydesign objects

**Usage**

```
ggbarweight_svy(surveyobj, x)
```

**Arguments**

surveyobj	svydesign
x	variable to plot

**Value**

ggplot object

**Examples**

```
library(survey)
data(api)
dstrat<-svydesign(id=~1,strata=~stype, weights=~pw, data=apistrat, fpc=~fpc)
ggbarweight_svy(dstrat, stype)+ylab("Proportion")
data(nhanes)
design <- svydesign(id=~SDMVPSU, strata=~SDMVSTRA, weights=~WTMEC2YR, nest=TRUE,data=nhanes)+ylab("Proportion")
ggbarweight_svy(design, race)
```

ggboxweight

*Weighted Box Plot of One Variable***Description**

Weighted Box Plot of One Variable

**Usage**

```
ggboxweight(df, x, weights)
```

**Arguments**

df	data frame
x	first variable of interest
weights	survey weights that sums to sample size

**Examples**

```
library(survey)
data(api)
ggboxweight(apistrat, api00, pw)
data(election)
ggboxweight(election_pps, Bush, p)
```

---

ggboxweight2d	<i>Weighted Boxplot with a categorical variable</i>
---------------	---

---

**Description**

Weighted Boxplot with a categorical variable

**Usage**

```
ggboxweight2d(df, x, y, weights)
```

**Arguments**

df	data frame
x	numeric variable of interest
y	categorical variable of interest
weights	survey weights that sums to sample size

**Examples**

```
library(survey)
data(api)
ggboxweight2d(apistrat, api00, stype, pw)
```

---

ggboxweight2d_svy	<i>Weighted Boxplot of a survey object with a categorical variable</i>
-------------------	--

---

**Description**

Weighted Boxplot of a survey object with a categorical variable

**Usage**

```
ggboxweight2d_svy(surveyobj, x, y)
```

**Arguments**

surveyobj	svy.design object
x	variable to boxplot
y	categorical variable

**Examples**

```
library(survey)
data(api)
dstrat<-svydesign(id=~1,strata=~stype, weights=~pw, data=apistrat, fpc=~fpc)
ggboxweight2d_svy(dstrat, api00, stype)
```

---

ggboxweight3d	<i>Weighted Boxplot with a categorical x axis and a faceting variable</i>
---------------	---

---

**Description**

Weighted Boxplot with a categorical x axis and a faceting variable

**Usage**

```
ggboxweight3d(df, x, y, z, weights)
```

**Arguments**

df	data frame
x	first categorical variable of interest
y	numeric variable of interest
z	second variable of interest for faceting
weights	survey weights that sums to sample size

**Examples**

```
library(survey)
data(api)
ggboxweight3d(apistrat, api00, stype,awards, pw)
```

---

ggboxweight3d_svy	<i>Weighted Boxplot of svy.design object with two categorical variables</i>
-------------------	---

---

**Description**

Weighted Boxplot of svy.design object with two categorical variables

**Usage**

```
ggboxweight3d_svy(surveyobj, x, y, z)
```

**Arguments**

surveyobj	svy.design
x	variable to boxplot
y	first categorical variable
z	second categorical variable (for faceting)

**Examples**

```
library(survey)
data(api)
dstrat<-svydesign(id=~1,strata=~stype, weights=~pw, data=apistrat, fpc=~fpc)
ggboxweight3d_svy(dstrat, api00, stype, awards)
```

---

ggboxweight_svy	<i>Weighted Box Plot of svy.design object</i>
-----------------	---

---

**Description**

Weighted Box Plot of svy.design object

**Usage**

```
ggboxweight_svy(surveyobj, x)
```

**Arguments**

surveyobj	svy.design object
x	variable to boxplot

**Value**

ggplot boxplot

**Examples**

```
library(survey)
data(api)
dstrat<-svydesign(id=~1,strata=~stype, weights=~pw, data=apistrat, fpc=~fpc)
ggboxweight_svy(dstrat, api00)
```



---

`gghexweight`*Weighted Hex Plot*

---

**Description**

Weighted Hex Plot

**Usage**

```
gghexweight(df, x, y, weights)
```

**Arguments**

<code>df</code>	data frame
<code>x</code>	name of variable for x axis
<code>y</code>	name of variable for y axis
<code>weights</code>	desired binwidth, if NULL bins in <code>geom_histogram</code> defaults to 30

**Examples**

```
data(api)
gghexweight(apistrat, api99, api00, pw)
```

---

`gghexweight2d`*Weighted Hex Plot with One Facet Variable*

---

**Description**

Weighted Hex Plot with One Facet Variable

**Usage**

```
gghexweight2d(df, x, y, z, weights)
```

**Arguments**

<code>df</code>	data frame
<code>x</code>	name of variable for x axis
<code>y</code>	name of variable for y axis
<code>z</code>	faceting categorical variable
<code>weights</code>	desired binwidth, if NULL bins in <code>geom_histogram</code> defaults to 30

**Examples**

```
data(api)
gghexweight2d(apistrat, api99, api00, stype, pw)
```

---

gghexweight2d_svy	<i>Weighted Hex Plot of svy.design with One Facet Variable</i>
-------------------	--

---

**Description**

Weighted Hex Plot of svy.design with One Facet Variable

**Usage**

```
gghexweight2d_svy(surveyobj, x, y, z)
```

**Arguments**

surveyobj	svy.design
x	variable for x axis
y	variable for y axis
z	faceting variable

**Examples**

```
library(survey)
data(api)
dstrat<-svydesign(id=~1,strata=~stype, weights=~pw, data=apistrat, fpc=~fpc)
gghexweight2d_svy(dstrat, api99, api00, stype)
```

---

gghexweight3d	<i>Weighted Box Plot with Two Facet Variables</i>
---------------	---

---

**Description**

Weighted Box Plot with Two Facet Variables

**Usage**

```
gghexweight3d(df, x, y, a, b, weights)
```

**Arguments**

df	data frame
x	name of variable for x axis
y	name of variable for y axis
a	first faceting variable
b	second faceting variable
weights	desired binwidth, if NULL bins in geom_histogram defaults to 30

**Examples**

```
data(api)
gghexweight3d(apistrat, api99, api00, stype, awards, pw)
```

---

gghexweight3d_svy	<i>Weighted Hex Plot of svy.design with Two Faceting Variables</i>
-------------------	--

---

**Description**

Weighted Hex Plot of svy.design with Two Faceting Variables

**Usage**

```
gghexweight3d_svy(surveyobj, x, y, a, b)
```

**Arguments**

surveyobj	svy.design
x	variable for x axis
y	variable for y axis
a	horizontal facetting variable
b	vertical facetting variable

**Examples**

```
library(survey)
data(api)
dstrat<-svydesign(id=~1,strata=~stype, weights=~pw, data=apistrat, fpc=~fpc)
gghexweight3d_svy(dstrat, api99, api00, stype, awards)
```

---

gghexweight_svy	<i>Weighted Hex Plot of Survey Design Object</i>
-----------------	--

---

**Description**

Weighted Hex Plot of Survey Design Object

**Usage**

```
gghexweight_svy(surveyobj, x, y)
```

**Arguments**

surveyobj	svy.design
x	name of variable for x axis
y	name of variable for y axis

**Examples**

```
library(survey)
data(api)
dstrat<-svydesign(id=~1,strata=~stype, weights=~pw, data=apistrat, fpc=~fpc)
gghexweight_svy(dstrat, api99, api00)
```

---

gghistweight	<i>Weighted Histogram</i>
--------------	---------------------------

---

**Description**

Weighted Histogram

**Usage**

```
gghistweight(df, x, weights, binwidth = NULL)
```

**Arguments**

df	data frame
x	variable of interest
weights	survey weights that sum to sample size
binwidth	desired binwidth, if NULL bins in geom_histogram defaults to 30

**Examples**

```
library(survey)
data(api)
gghistweight(apistrat, api00, pw)
gghistweight(apistrat, api00, pw, binwidth = 10)
data(election)
gghistweight(election_pps, Bush, p)
```

---

gghistweight2d	<i>Weighted Histogram with One Facet</i>
----------------	--

---

**Description**

Weighted Histogram with One Facet

**Usage**

```
gghistweight2d(df, x, y, weights, binwidth = NULL)
```

**Arguments**

df	data frame
x	first variable of interest
y	categorical variable for faceting
weights	survey weights that sum to sample size
binwidth	desired binwidth, if NULL bins in geom_histogram defaults to 30

**Examples**

```
library(survey)
data(api)
gghistweight2d(apistrat, api00, stype, pw)
gghistweight2d(apistrat, api00, stype, pw, binwidth = 10)
```

---

gghistweight2d_svy	<i>Histogram of svy.object with One Facet</i>
--------------------	---

---

**Description**

Histogram of svy.object with One Facet

**Usage**

```
gghistweight2d_svy(surveyobj, x, y, binwidth = NULL)
```

**Arguments**

surveyobj	svy.design object
x	variable to histogram
y	categorical variable to facet
binwidth	binwidth to pass to geom_hist

**Examples**

```
library(survey)
data(api)
dstrat<-svydesign(id=~1,strata=~stype, weights=~pw, data=apistrat, fpc=~fpc)
gghistweight2d_svy(dstrat, api00, stype)
gghistweight2d_svy(dstrat, api00, stype, binwidth = 10)
```

---

gghistweight3d	<i>Weighted Histogram with Two Facets</i>
----------------	---

---

**Description**

Weighted Histogram with Two Facets

**Usage**

```
gghistweight3d(df, x, y, z, weights, binwidth = NULL)
```

**Arguments**

df	data frame
x	first variable of interest
y	first categorical variable for faceting
z	second categorical variable for faceting
weights	survey weights that sum to sample size
binwidth	desired binwidth, if NULL bins in geom_histogram defaults to 30

**Examples**

```
library(survey)
data(api)
gghistweight3d(apistrat, api00, stype, awards, pw)
gghistweight3d(apistrat, api00, stype, awards, pw, binwidth = 10)
```

---

gghistweight3d_svy	<i>Histogram of svy.design object with two facets</i>
--------------------	---

---

**Description**

Histogram of svy.design object with two facets

**Usage**

```
gghistweight3d_svy(surveyobj, x, y, z, binwidth = NULL)
```

**Arguments**

surveyobj	svy.design object
x	variable to histogram
y	horizontal facet
z	vertical facet
binwidth	binwidth to pass to geom_hist

**Examples**

```
library(survey)
data(api)
dstrat<-svydesign(id=~1,strata=~stype, weights=~pw, data=apistrat, fpc=~fpc)
gghistweight3d_svy(dstrat, api00, stype, awards)
gghistweight3d_svy(dstrat, api00, stype, awards, binwidth = 10)
```

---

gghistweight_svy	<i>Histogram of svgsdesign object</i>
------------------	---------------------------------------

---

**Description**

Histogram of svgsdesign object

**Usage**

```
gghistweight_svy(surveyobj, x, binwidth = NULL)
```

**Arguments**

surveyobj	svy.design object
x	variable to histogram
binwidth	binwidth to pass to geom_hist

**Examples**

```
library(survey)
data(api)
dstrat<-svydesign(id=~1,strata=~stype, weights=~pw, data=apistrat, fpc=~fpc)
gghistweight_svy(dstrat, api00)
gghistweight_svy(dstrat, api00, binwidth = 10)
```

---

hello	<i>Hello, World!</i>
-------	----------------------

---

**Description**

Prints 'Hello, world!'.

**Usage**

```
hello()
```

**Examples**

```
hello()
```

# Index

ggbarcrosstabs, [2](#)  
ggbarcrosstabs3d, [2](#)  
ggbarcrosstabs3d\_svy, [3](#)  
ggbarcrosstabs\_svy, [3](#)  
ggbarweight, [4](#)  
ggbarweight\_svy, [4](#)  
ggboxweight, [5](#)  
ggboxweight2d, [6](#)  
ggboxweight2d\_svy, [6](#)  
ggboxweight3d, [7](#)  
ggboxweight3d\_svy, [7](#)  
ggboxweight\_svy, [8](#)  
gghexweight, [8](#)  
gghexweight2d, [9](#)  
gghexweight2d\_svy, [9](#)  
gghexweight3d, [10](#)  
gghexweight3d\_svy, [10](#)  
gghexweight\_svy, [11](#)  
gghistweight, [11](#)  
gghistweight2d, [12](#)  
gghistweight2d\_svy, [12](#)  
gghistweight3d, [13](#)  
gghistweight3d\_svy, [13](#)  
gghistweight\_svy, [14](#)  
  
hello, [14](#)