

# An overview of the plot and stats functions in the **surveyor** package

Andrie de Vries

July 26, 2012

**surveyor** is a package that makes it easy to create graphical crosstabs from survey data files.

## 1 Introduction

blah

## 2 Create test data and a surveyor object

```
> library(surveydata)
> library(surveyor)
> library(ggplot2)
> library(lattice)
> qData <- data.frame(
+   Q1 = c("Yes", "No", "Yes", "No", "Yes"),
+   Q2 = c(1, 1, 1, 0, 0),
+   Q3 = c("A", "B", "C", "B", "A"),
+   Q4_1 = c("Yes", "No", "Yes", "No", "Yes"),
+   Q4_2 = c("Yes", "Yes", "No", "No", "Yes"),
+   Q5_1 = c(1, 1, 1, 0, 0),
+   Q5_2 = c(0, 0, 1, 1, 1),
+   Q6_1 = c("A", "B", "C", "B", "A"),
+   Q6_2 = c("C", "B", "A", "B", "C"),
+   crossbreak = c("AAA", "AAA", "BBB", "BBB", "BBB"),
+   crossbreak2 = c("DDD", "EEE", "DDD", "EEE", "DDD"),
+   weight = c(0.9, 1.1, 0.8, 1.2, 1.0)
+ )
> varlabels(qData) <- c(
```

```

+   "Single question with yes/no input",
+   "Single question with binary input",
+   "Single question with multiple options",
+   "Multiple question with yes/no input: XX",
+   "Multiple question with yes/no input: YY",
+   "Multiple question with binary input: XX",
+   "Multiple question with binary input: YY",
+   "Multiple question with multiple options: XX",
+   "Multiple question with multiple options: YY",
+   "crossbreak",
+   "crossbreak2",
+   "weight")
> qData <- as.surveydata(qData, renameVarlabels=TRUE)

```

Next, set up a surveyor object. For the sake of illustrating the different options, we create two surveyor objects:

- sg: surveyor object for ggplot graphics
- sl: surveyor object for lattice graphics

```

> s <- as.surveyor(qData, crossbreak=list(groups=qData$crossbreak), qData$weight)
> sg <- surveyorUpdateDefaults(s, addPlotTitle = TRUE)
> sl <- surveyorUpdateDefaults(sg, fastgraphics=TRUE)
>

```

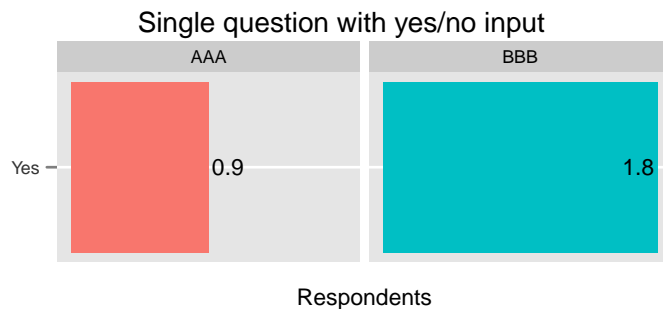
### 3 Demonstrating the different plotBar options

#### 3.1 bar plots with ggplot

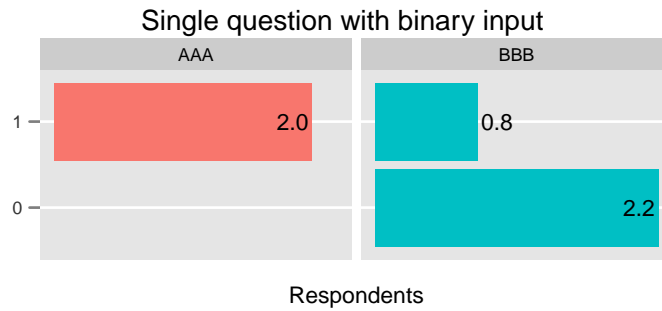
```

> x <- plotBar(statsBin(codeGuess(sg, "Q1")))
> print(x$plot)

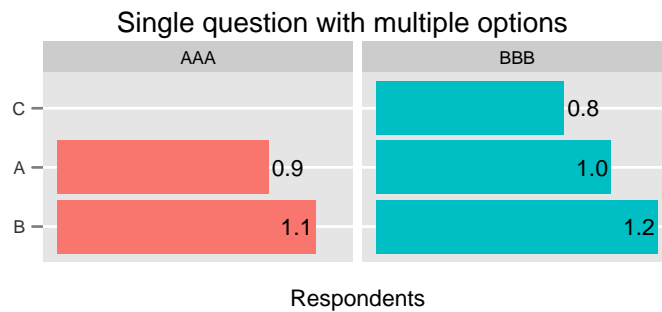
```



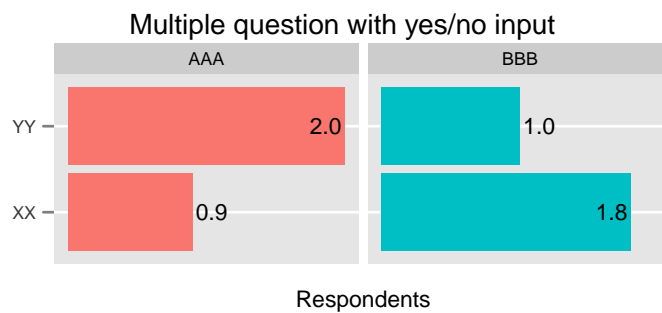
```
> x <- plotBar(statsBin(codeGuess(sg, "Q2")))
> print(x$plot)
```



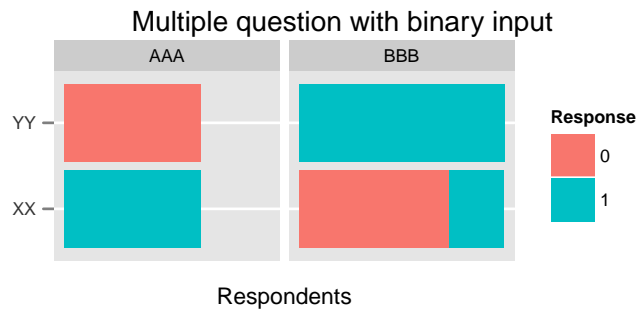
```
> x <- plotBar(statsBin(codeGuess(sg, "Q3")))
> print(x$plot)
```



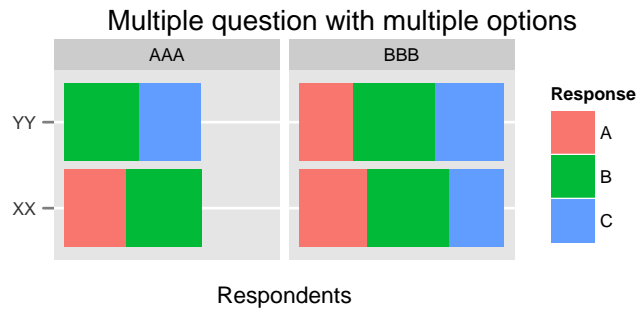
```
> x <- plotBar(statsBin(codeGuess(sg, "Q4")))
> print(x$plot)
```



```
> x <- plotBar(statsBin(codeGuess(sg, "Q5")))
> print(x$plot)
```



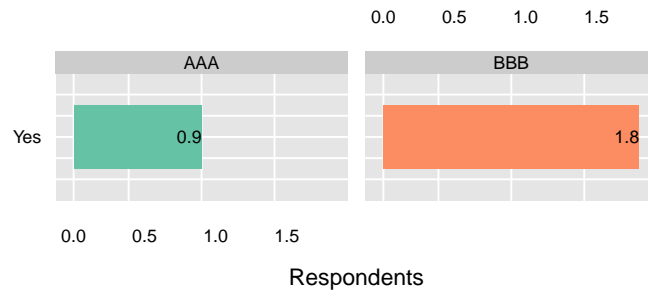
```
> x <- plotBar(statsBin(codeGuess(sg, "Q6")))
> print(x$plot)
```



### 3.2 bar plots with lattice

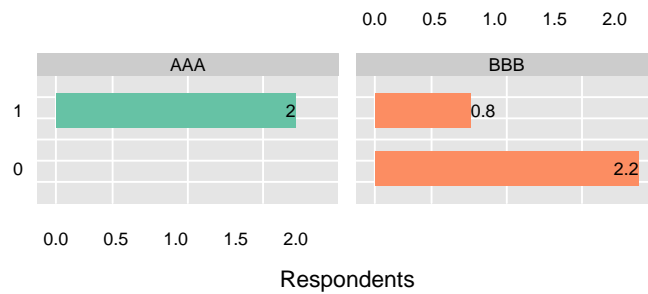
```
> x <- plotBar(statsBin(codeGuess(sl, "Q1")))
> print(x$plot)
```

### Single question with yes/no input



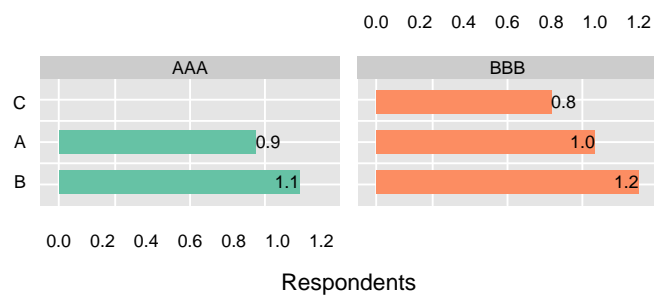
```
> x <- plotBar(statsBin(codeGuess(sl, "Q2")))
> print(x$plot)
```

### Single question with binary input



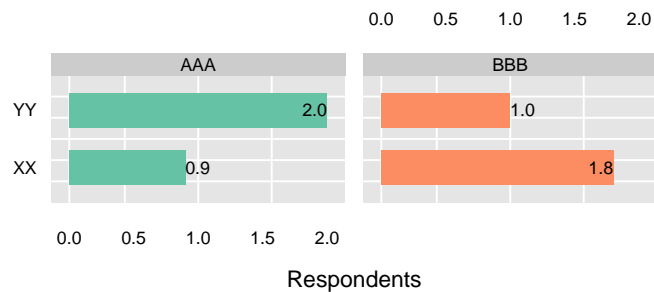
```
> x <- plotBar(statsBin(codeGuess(sl, "Q3")))
> print(x$plot)
```

### Single question with multiple options



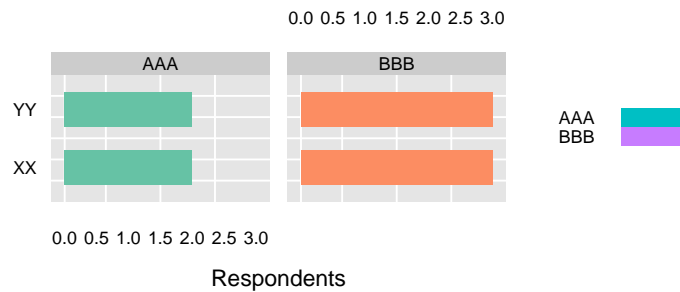
```
> x <- plotBar(statsBin(codeGuess(sl, "Q4")))
> print(x$plot)
```

### Multiple question with yes/no input



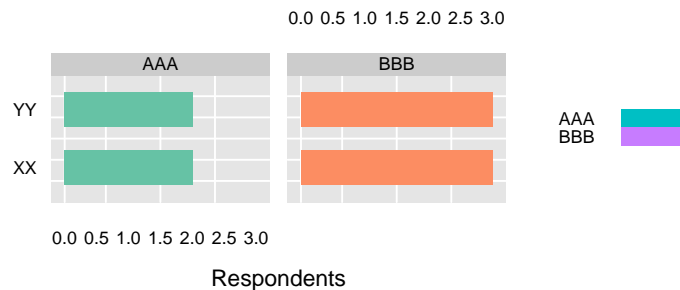
```
> x <- plotBar(statsBin(codeGuess(sl, "Q5")))
> print(x$plot)
```

### Multiple question with binary input



```
> x <- plotBar(statsBin(codeGuess(sl, "Q6")))
> print(x$plot)
```

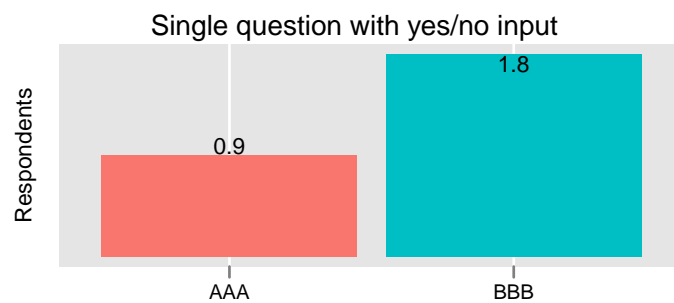
### Multiple question with multiple options



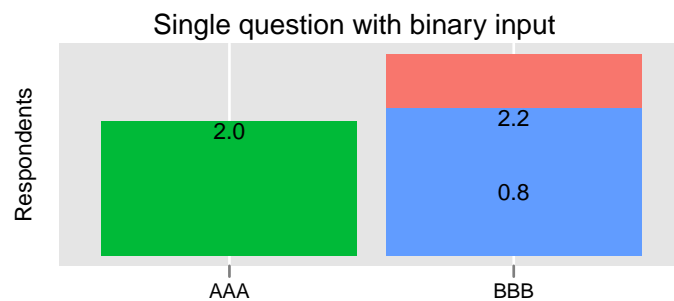
## 4 Demonstrating the different plotColumn options

### 4.1 Column plots with ggplot

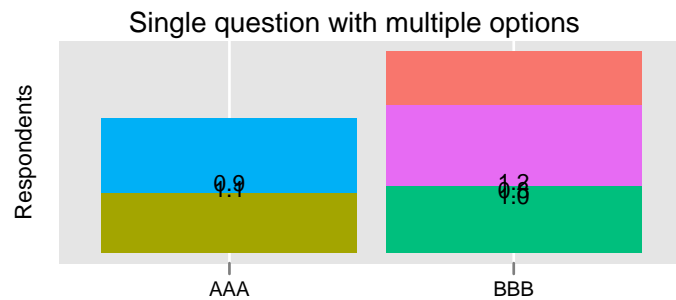
```
> x <- plotColumn(statsBin(codeGuess(sg, "Q1")))
> print(x$plot)
```



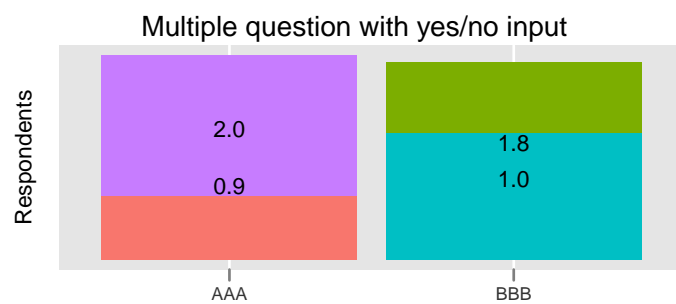
```
> x <- plotColumn(statsBin(codeGuess(sg, "Q2")))
> print(x$plot)
```



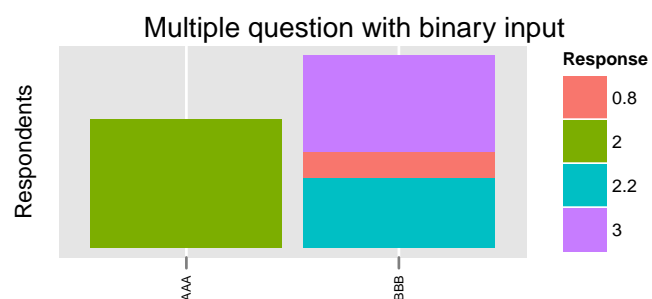
```
> x <- plotColumn(statsBin(codeGuess(sg, "Q3")))
> print(x$plot)
```



```
> x <- plotColumn(statsBin(codeGuess(sg, "Q4")))
> print(x$plot)
```

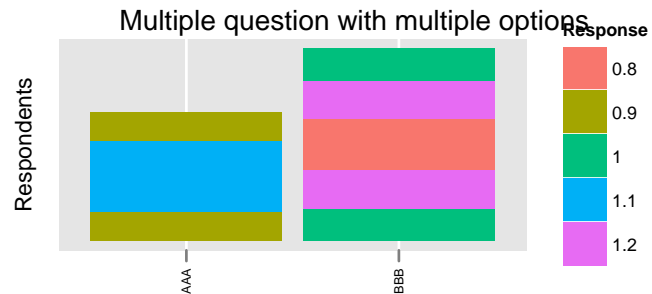


```
> x <- plotColumn(statsBin(codeGuess(sg, "Q5")))
> print(x$plot)
```



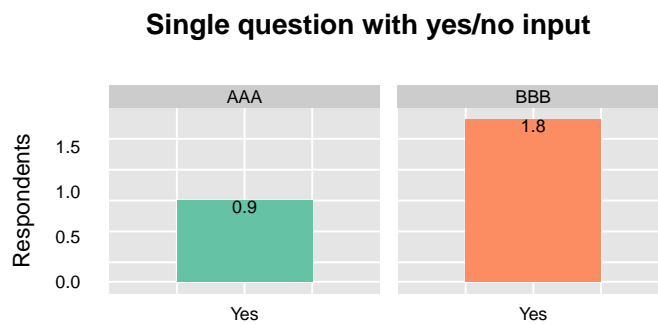
```
> x <- plotColumn(statsBin(codeGuess(sg, "Q6")))
> print(x$plot)
```



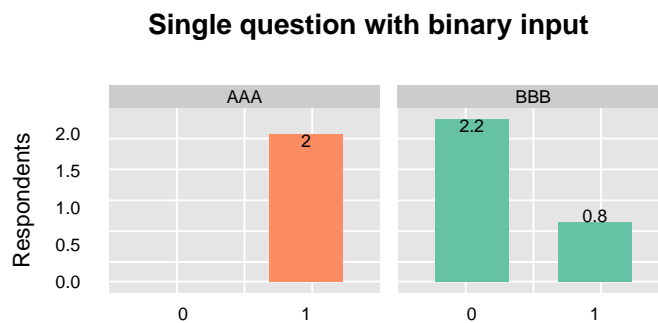


## 4.2 Column plots with lattice

```
> x <- plotColumn(statsBin(codeGuess(sl, "Q1")))
> print(x$plot)
```

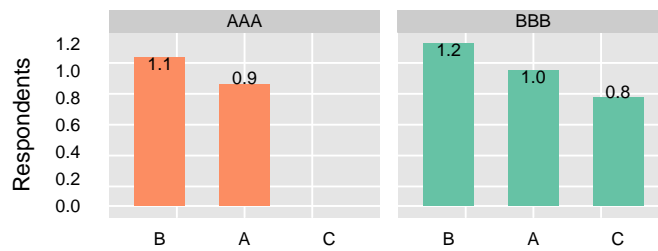


```
> x <- plotColumn(statsBin(codeGuess(sl, "Q2")))
> print(x$plot)
```



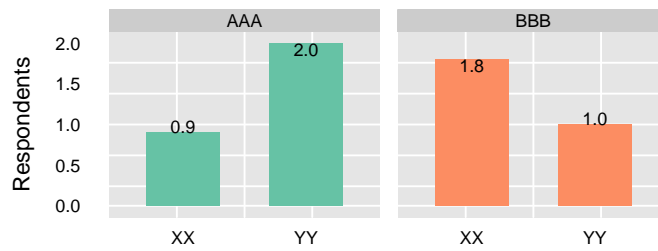
```
> x <- plotColumn(statsBin(codeGuess(sl, "Q3")))
> print(x$plot)
```

### Single question with multiple options



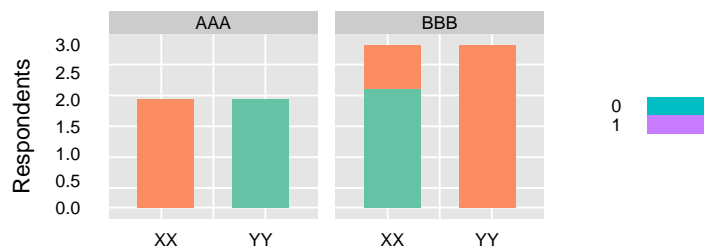
```
> x <- plotColumn(statsBin(codeGuess(sl, "Q4")))
> print(x$plot)
```

### Multiple question with yes/no input

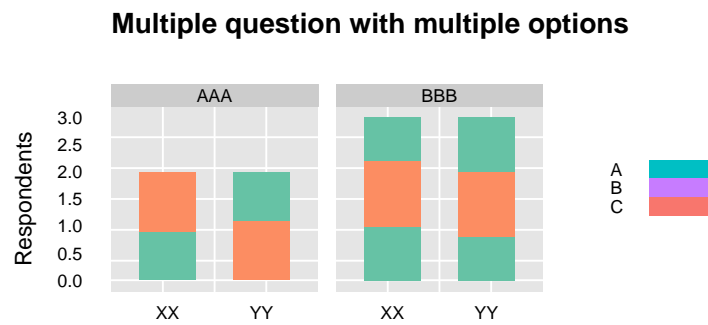


```
> x <- plotColumn(statsBin(codeGuess(sl, "Q5")))
> print(x$plot)
```

### Multiple question with binary input



```
> x <- plotColumn(statsBin(codeGuess(s1, "Q6")))
> print(x$plot)
```



## 5 Conclusion

The `surveyor` packages makes it easy to analyse survey data files.