

CUNY DATA 621 - Business Analytics and Data Mining

Mental Health Survey - Data Prep and Basic EDA

Group 1, 2018

```
if (!require('countrycode')) (install.packages('countrycode'))
if (!require('dplyr')) (install.packages('dplyr'))
if (!require('psych')) (install.packages('psych'))
if (!require('DataExplorer')) (install.packages('DataExplorer'))
if (!require('lubridate')) (install.packages('lubridate'))

url <- paste("https://raw.githubusercontent.com/",
  "RaphaelNash/CUNY_DATA621_GroupProject/master/Data/survey_RAW.csv",
  sep="")
df <- read.csv(url)
```

Data Source

The data here is taken from <https://www.kaggle.com/osmi/mental-health-in-tech-survey>, which in turn was generated from the Open Sourcing Mental Illness project (OSMI), which describes itself as “a non-profit, 501(c)(3) corporation dedicated to raising awareness, educating, and providing resources to support mental wellness in the tech and open source communities”. OSMI conducted survey-based research in 2014, 2016, and 2017, to collect data about professionals in the open source developer community and research their openness to accessing mental health resources. The data we’re using in this analysis comes from the 2014 survey results.

Basic Statistics

The data is 307.5 Kb in size. There are 1,259 rows and 27 columns (features). Of all 27 columns, 26 are discrete, 1 are continuous, and 0 are all missing. There are 1,892 missing values out of 33,993 data points.

DATA PREPARATION

Comments

This field is fascinating and ripe for text analysis. We’ll leave it in for now, but it would need to be further prepared for any regression to be done.

```
sample <- df[!(is.na(df$comments)), ]
head(sample$comments)
```

[1] I’m not on my company’s health insurance which could be part of the reason I answered Don’t know to so many questions.

[2] I have chronic low-level neurological issues that have mental health side effects. One of my supervisors has also experienced similar neurological problems so I feel more comfortable being open about my issues than I would with someone without that experience.

[3] My company does provide healthcare but not to me as I’m on a fixed-term contract. The mental healthcare I use is provided entirely outside of my work.

[4] Relatively new job. Ask again later

[5] Sometimes I think about using drugs for my mental health issues. If i use drugs I feel better
[6] I selected my current employer based on its policies about self care and the quality of their overall health and wellness benefits. I still have residual caution from previous employers who ranged from ambivalent to indifferent to actively hostile regarding mental health concerns. 160 Levels: ... you rock for doing this!

Date

It's likely that the timestamp field will be omitted entirely from analysis, but should someone wish to use it, we'll convert it to the appropriate type.

```
df$Timestamp <- ymd_hms(df$Timestamp)
```

Age

Our Age variable has some clear and impossible outliers. There are multiple values < 18 (even some negative numbers) and some values > 200 years old. Instead of replacing these, for now, let's set to NA and impute later.

```
df$Age[df$Age > 73 | df$Age < 18 ] <- NA
```

Gender

Gender is more complex in this dataset. Let's start by doing some rough matching and cleaning.

```
df$Gender <- tolower(df$Gender)
df$Gender <- trimws(df$Gender)

### start with the obvious
cis_female_syn <- c("femail", "f", "woman", "femail", "female (cis)",
                  "cis female", "cis-female/femme", "femake",
                  "female")
df$Gender[df$Gender %in% cis_female_syn] <- "female_cis"

cis_male_syn <- c("m", "man", "male (cis)", "male", "mal", "mail",
                "maile", "cis man", "cis male", "msle", "malr",
                "make")
df$Gender[df$Gender %in% cis_male_syn] <- "male_cis"

trans_female_syn <- c("trans woman", "trans-female", "female (trans)")
df$Gender[df$Gender %in% trans_female_syn] <- "female_trans"

genderqueer_syn <- c("non-binary", "enby", "queer", "queer/she/they",
                  "fluid", "androgynous", "agender", "neuter")
df$Gender[df$Gender %in% genderqueer_syn] <- "genderqueer"

fluid_syn <- c("male leaning androgynous", "male-ish",
              "ostensibly male, unsure what that really means",
              "something kinda male?", "guy (-ish) ^_^")
df$Gender[df$Gender %in% fluid_syn] <- "fluid"

unknown <- c("a little about you", "all", "p", "nah")
df$Gender[df$Gender %in% unknown] <- "unknown"
```

```
### Let's update some call out issues.  Obvs 967 reported "female"
# in the Gender field, but noted being a trans woman in the comments.
df$Gender[967] <- "female_trans"

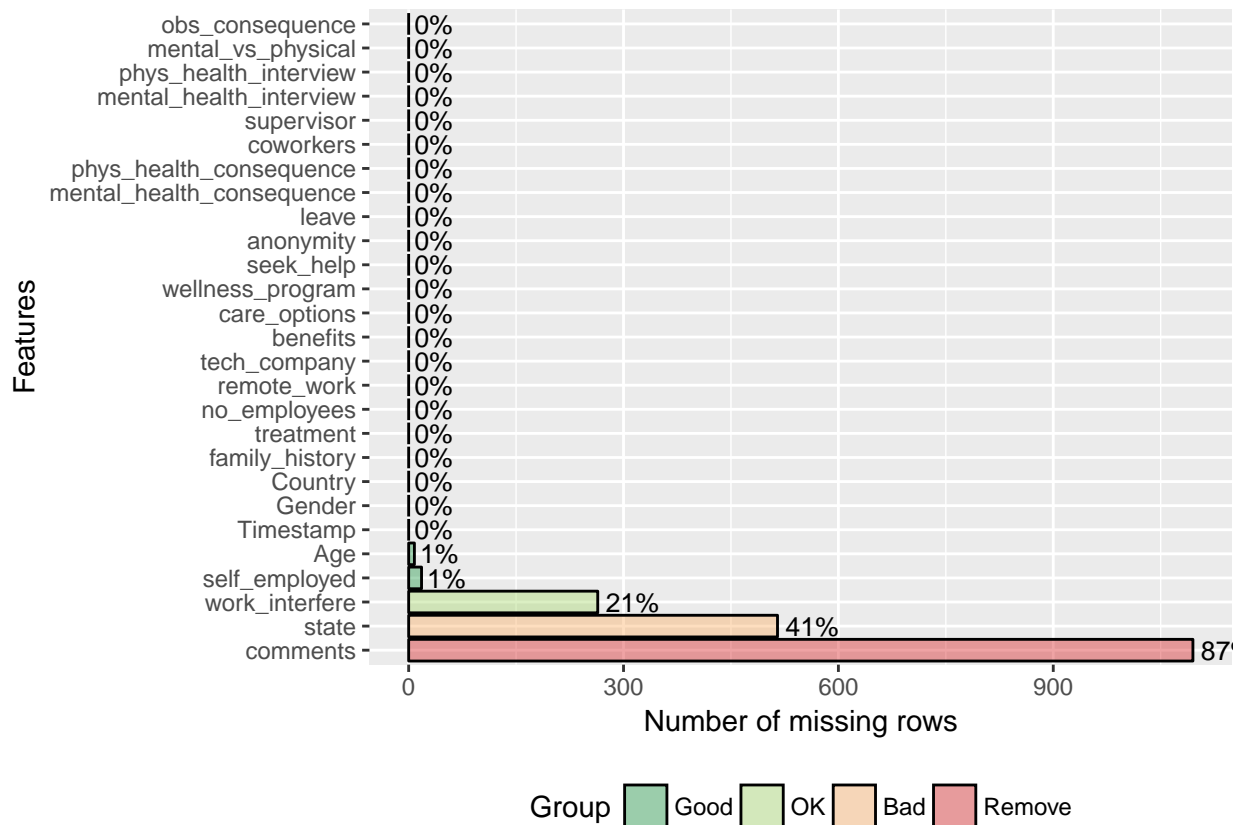
df$Gender <- as.factor(df$Gender)
table(df$Gender)
```

```
##
##   female_cis female_trans      fluid  genderqueer   male_cis
##         246          5          5          9        990
##   unknown
##         4
```

Missing

Plot and Review Missing

```
plot_missing(df)
```



State

Most of our missing values are for US States. While it's fine for this to be missing if it's a non-US country, let's make sure that's all that's happening.

```
#Number of observations that aren't United States
nrow(df[df$Country != "United States",])
```

```
## [1] 508
#Number of missing states
sum(is.na(df$state))

## [1] 515
nrow(df[df$Country == "United States" & is.na(df$state),])

## [1] 11
# there are 11 missing states.
df$state <- as.character(df$state)
df$state[df$Country == "United States" & is.na(df$state)] <- "Unknown"

# Still some missing: non-US countries w/ states?!
sub <- df[df$Country != "United States" & !is.na(df$state),]

knitr::kable(sub[, c("state", "Country")])
```

	state	Country
320	NY	Latvia
489	MD	Israel
990	IL	Bahamas, The
1180	UT	Bulgaria

```
# Ok, that's weird. Let's NA those
df$state[df$Country != "United States" & !is.na(df$state)] <- NA

df$state <- as.factor(df$state)
rm(sub)
```

Feature Creation: Continent

Since country and state are proving to be non-uniform, let's use the great country code package to create a "continent" feature that may be useful for regression.

```
df$continent <- as.factor(countrycode(sourcevar = df[, "Country"],
                                     origin = "country.name",
                                     destination = "continent"))

table(df$continent)
```

```
##
## Africa Americas Asia Europe Oceania
##      8      837    24    361    29
```

Work Interfere | Self Employed | Age

```
summary(df[, c("work_interfere", "self_employed", "Age")])

## work_interfere self_employed Age
## Never :213 No :1095 Min. :18.00
```

```
## Often      :144      Yes : 146      1st Qu.:27.00
## Rarely     :173      NA's:  18      Median :31.00
## Sometimes:465                      Mean   :32.08
## NA's       :264                      3rd Qu.:36.00
##                               Max.    :72.00
##                               NA's    :8
```

The work_interfere variable is a response to the question: “If you have a mental health condition, do you feel that it interferes with your work?” I would suggest two possible interpretations:

- 1) “I do not have a mental health condition”
- 2) “I don’t want to respond about how my work is affected”

Since the treatment variable is pretty evenly split (No=622/Yes=637) on whether they’ve sought treatment for a mental health condition, it may not always be option 1. Since we have no way of knowing which condition is likely, let’s simply add a 5th category for “No Response”

```
df$work_interfere <- as.character(df$work_interfere)
df$work_interfere[is.na(df$work_interfere)] <- "No Response"
df$work_interfere <- as.factor(df$work_interfere)
```

For the remaining two fields, since we will only lose 26 observations, let’s simply remove those observations.

```
df <- df[!is.na(df$self_employed),]
df <- df[!is.na(df$Age),]
```

EDA

With this, our data is roughly ready for review. We still have NAs in State and Comments, but if used, both would need to be handled carefully in other ways (eg - State, but subsetting to only US data, Comments to craft some NLP features). The Timestamp variable may also be easily dropped.

Data Summary

```
summary <- describe(df[,c(2:26, 28)]),c(2:5,8,9,11,12))
knitr::kable(summary)
```

	n	mean	sd	median	min	max	skew	kurtosis
Age	1233	32.061638	7.3053372	31	18	72	1.0190819	1.8785954
Gender*	1233	4.191403	1.5926039	5	1	6	-1.4807195	0.2212752
Country*	1233	38.831306	13.2436861	46	1	47	-1.6255591	1.0946610
state*	735	23.068027	14.6803637	27	1	46	-0.1375923	-1.4753130
self_employed*	1233	1.115166	0.3193520	1	1	2	2.4081366	3.8022074
family_history*	1233	1.390916	0.4881537	1	1	2	0.4465604	-1.8020433
treatment*	1233	1.505272	0.5001751	2	1	2	-0.0210623	-2.0011774
work_interfere*	1233	3.332522	1.5467986	4	1	5	-0.2344497	-1.5134693
no_employees*	1233	3.795620	1.7391819	4	1	6	-0.1622824	-1.3478810
remote_work*	1233	1.296026	0.4566878	1	1	2	0.8925512	-1.2043271
tech_company*	1233	1.818329	0.3857294	2	1	2	-1.6491931	0.7204242
benefits*	1233	2.053528	0.8366926	2	1	3	-0.1006465	-1.5656502
care_options*	1233	1.952960	0.8656840	2	1	3	0.0905348	-1.6607894
wellness_program*	1233	2.036496	0.5750195	2	1	3	0.0014455	0.0095658
seek_help*	1233	1.911598	0.6928228	2	1	3	0.1187330	-0.9217245

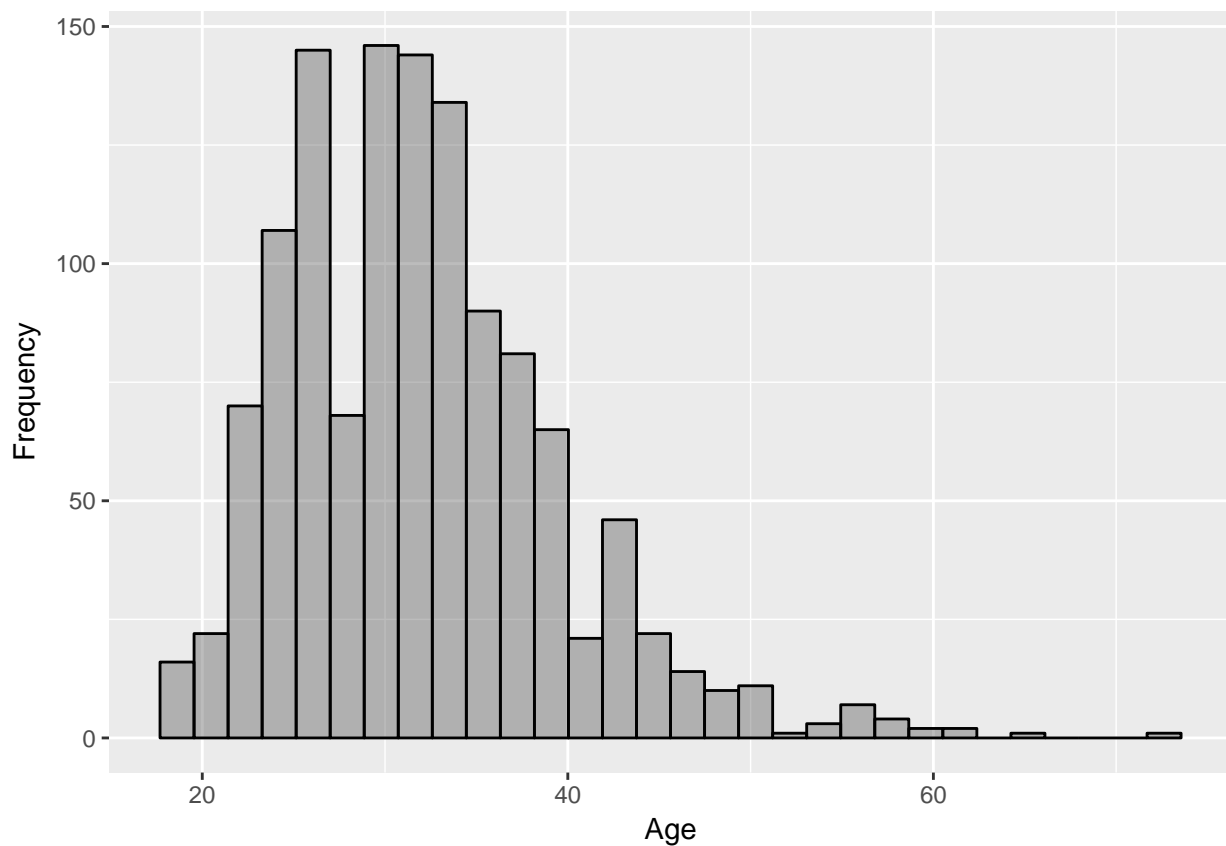
	n	mean	sd	median	min	max	skew	kurtosis
anonymity*	1233	1.649635	0.9101887	1	1	3	0.7462613	-1.3762742
leave*	1233	2.412003	1.5093600	2	1	5	0.5605637	-1.1402516
mental_health_consequence*	1233	1.849959	0.7693618	2	1	3	0.2625304	-1.2723555
phys_health_consequence*	1233	1.824817	0.4853040	2	1	3	-0.4035896	0.4148329
coworkers*	1233	1.965937	0.6167108	2	1	3	0.0205104	-0.3785987
supervisor*	1233	2.094079	0.8427916	2	1	3	-0.1788474	-1.5707405
mental_health_interview*	1233	1.867802	0.4201226	2	1	3	-0.8083882	1.6011142
phys_health_interview*	1233	1.715329	0.7224824	2	1	3	0.4874161	-0.9749955
mental_vs_physical*	1233	1.813463	0.8350560	2	1	3	0.3604057	-1.4751512
obs_consequence*	1233	1.145174	0.3524195	1	1	2	2.0120218	2.0498962
continent*	1233	2.663422	0.9775567	2	1	5	0.8271968	-1.0322071

Histogram of Variables

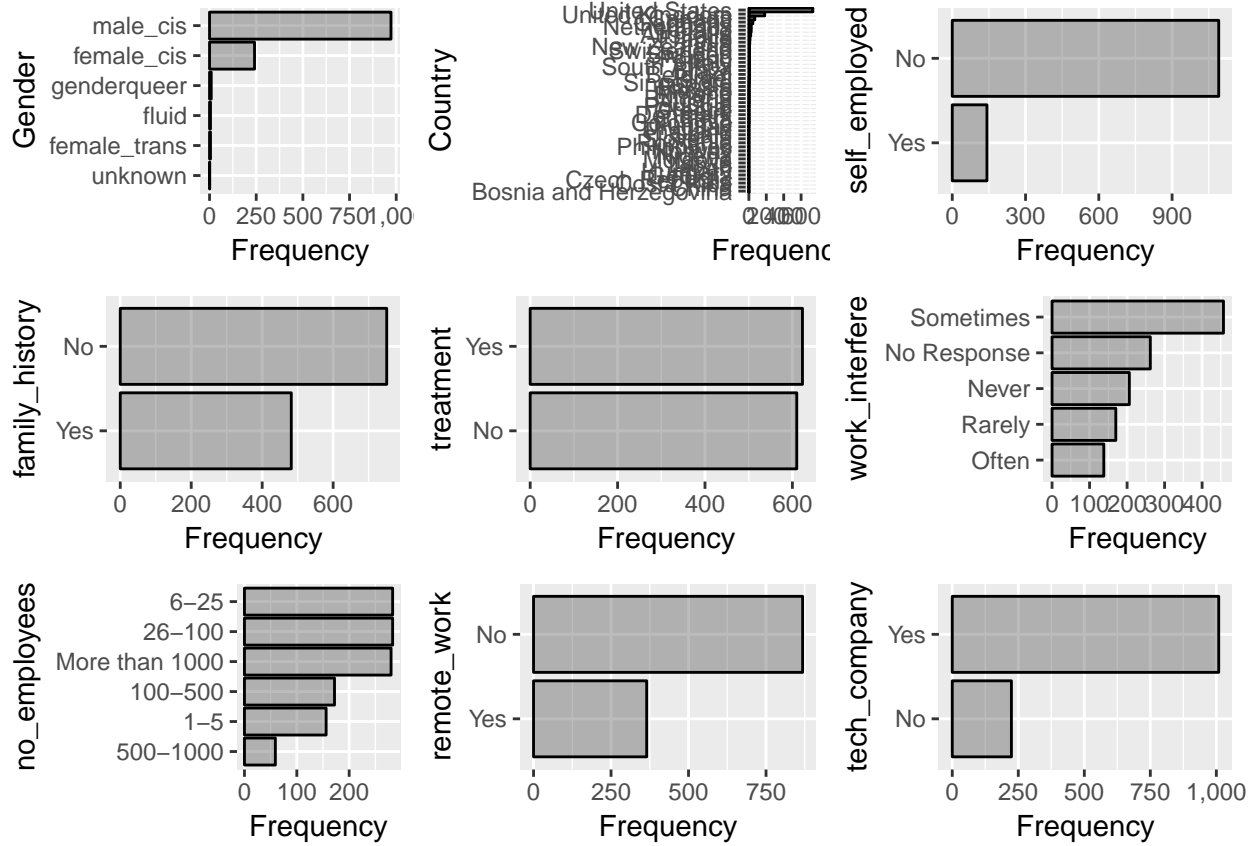
```
clean <- df
clean$Timestamp <- NULL
clean$comments <- NULL
clean$state <- NULL

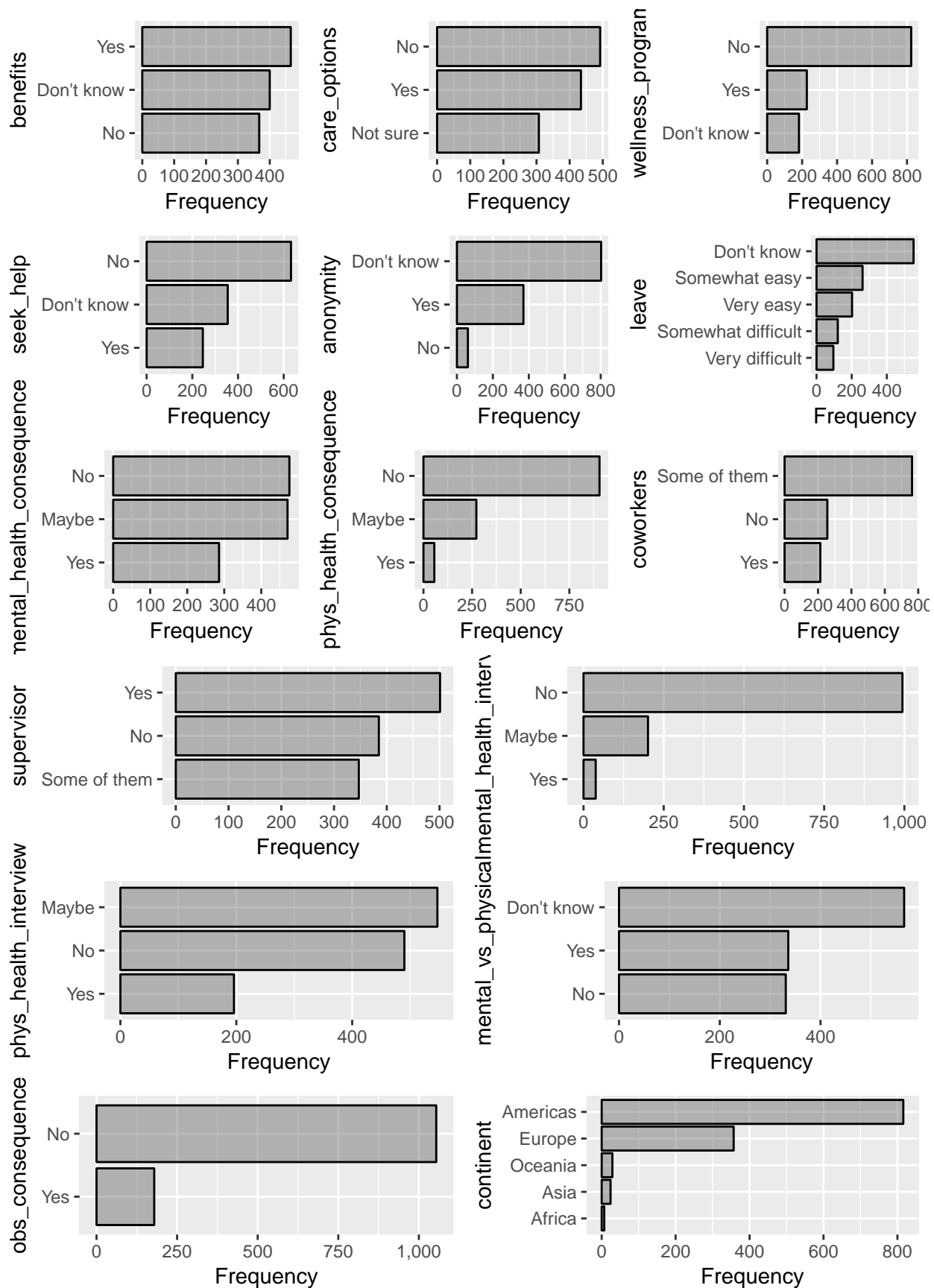
out <- split_columns(clean)

plot_histogram(out$continuous)
```



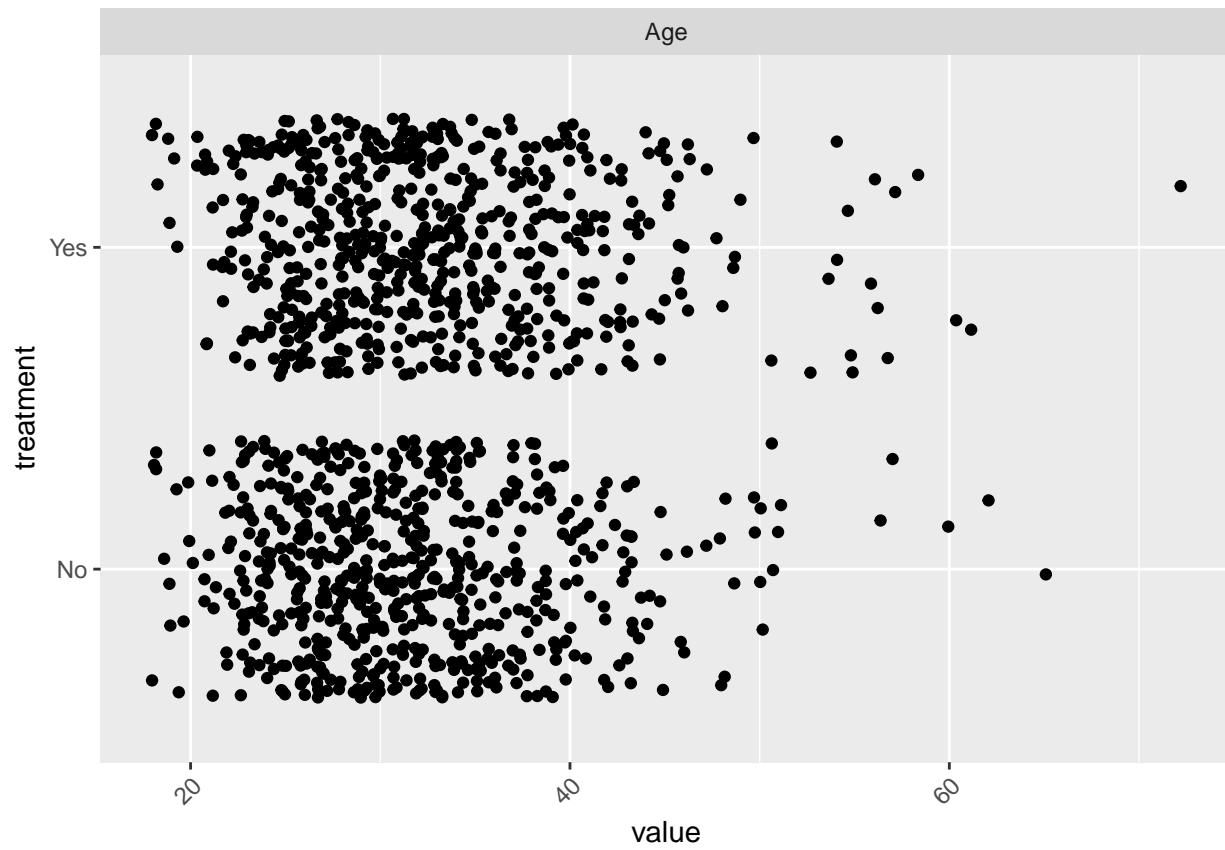
```
plot_bar(out$discrete)
```

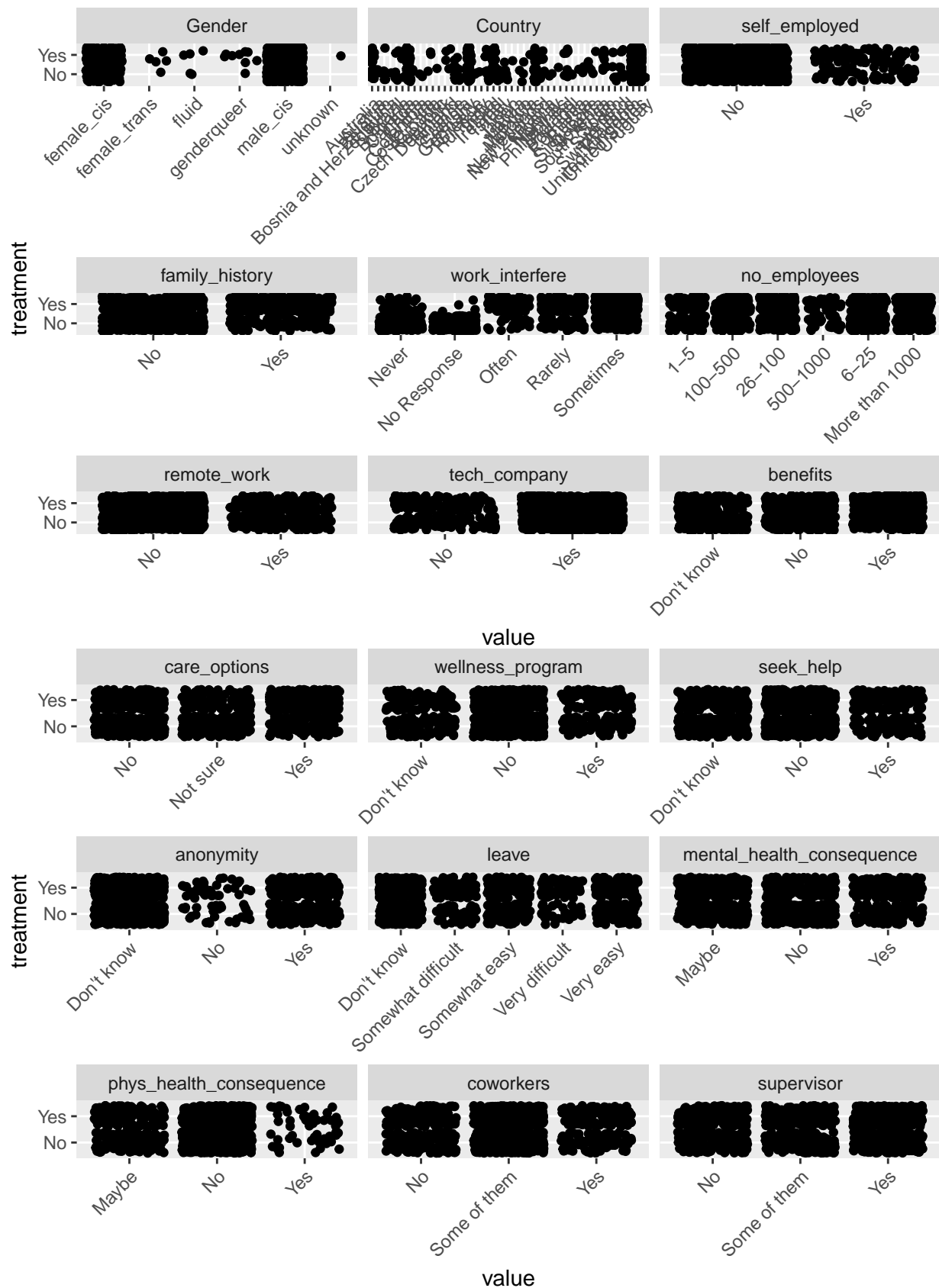


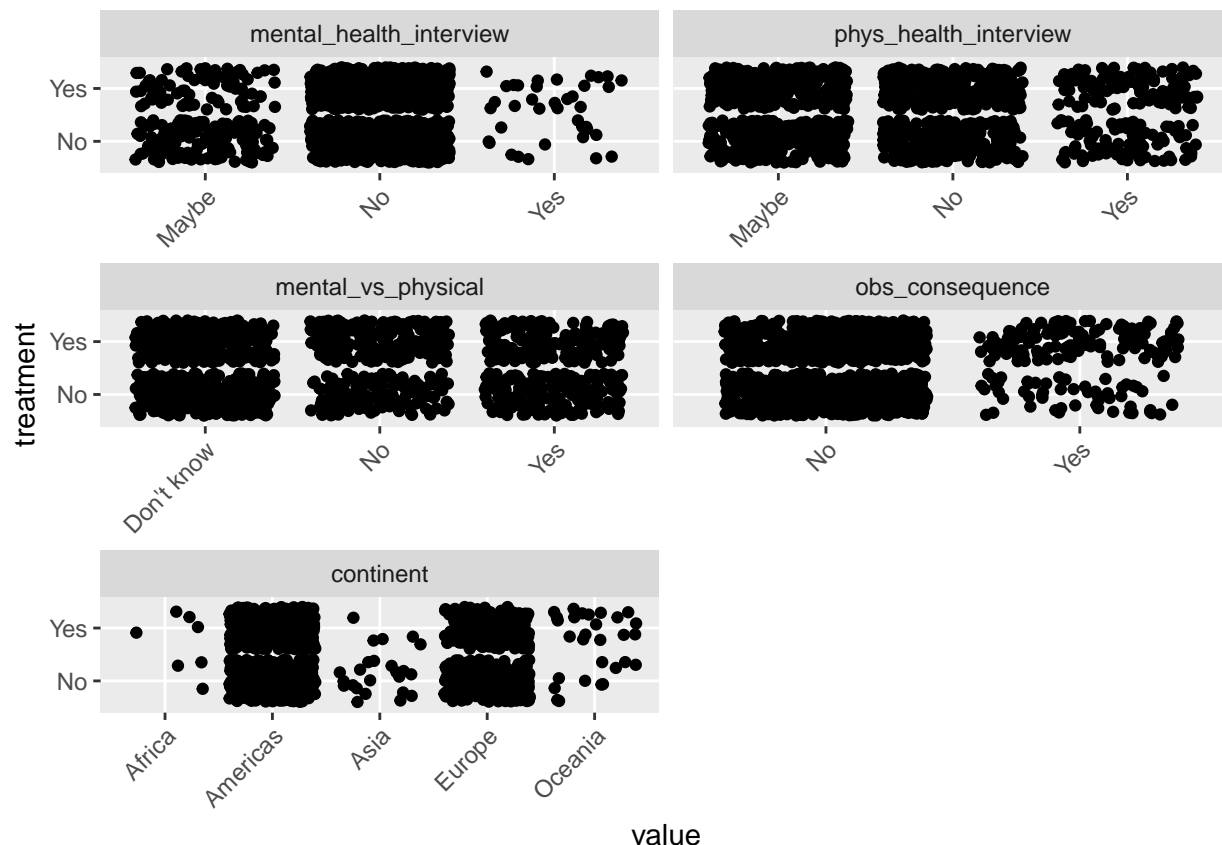


Relationship of Predictors to Target: “treatment”

```
plot_scatterplot(clean, "treatment", position = "jitter")
```







Joy's Version 1 of Relationship of Predictors to Target:

This is an alternative look that eliminates the overplotting...

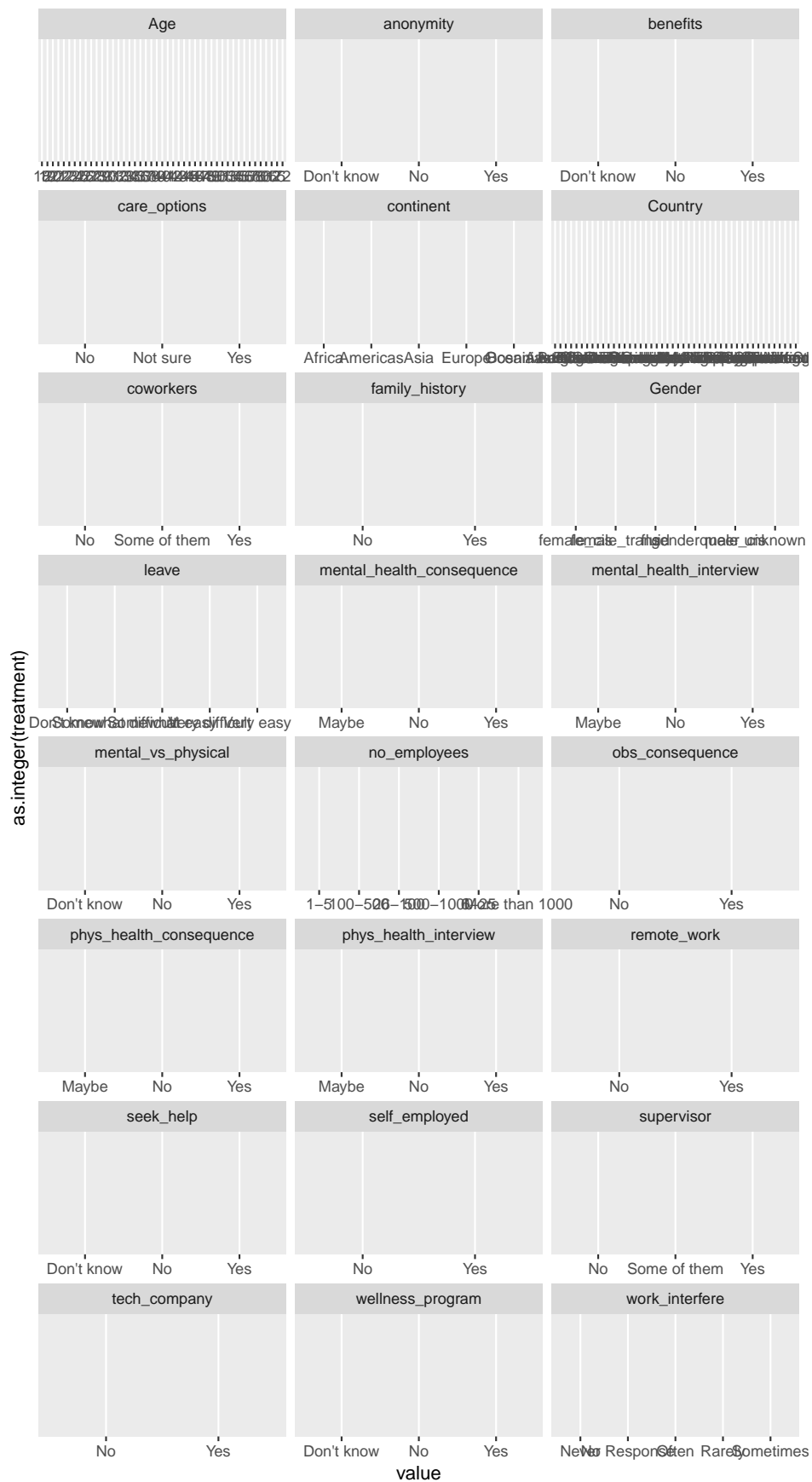
```
library(tidyr)
library(ggplot2)

##
## Attaching package: 'ggplot2'
## The following objects are masked from 'package:psych':
##
##      %+%, alpha
clean_long <- clean %>% gather(-treatment, key = "key", value = "value", convert = TRUE)

## Warning: attributes are not identical across measure variables;
## they will be dropped

ggplot(data = clean_long, aes(x=value, y=as.integer(treatment))) +
  geom_smooth() +
  #geom_point(alpha=0.4) +
  facet_wrap( ~ key, scales = "free", ncol = 3)

## `geom_smooth()` using method = 'gam'
```



Cleanup and Save

```
saveRDS(df, "../Data/MentalHealthCLEAN.rds")
```