Informal Rust Gamedev in 2024 Survey

Jan Hohenheim

2024-05-04

Dependencies

```
library(tidyverse)
library(glue)
library(ggthemes)
library(latex2exp)
library(glmnet)
library(MASS)
library(rstatix)
library(ordinal)
library(MASS)
library(DescTools)
library(wordcloud2)
library(webshot)
library("htmlwidgets")
webshot::install_phantomjs(force = FALSE)

theme_set(theme_solarized_2())
palette <- function(n) {
  scale_color_solarized()$palette(n)[[n]]
}
```

Data Wrangling

Data Cleaning

```
dat_raw <- read_csv("data_original.csv")
```

```
## Rows: 52 Columns: 12
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (10): Timestamp, How would you change the amount of content per newslett...
## dbl  (2): On a scale of 1 to 5, how high is your excitement for the newslett...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
dat <- dat_raw |>
  # Change names
  ## Timestamp
  rename_at(1, ~ "timestamp") |>
  ## On a scale of 1 to 5, how high is your excitement for the newsletter in its current form?
  rename_at(2, ~ "excitement") |>
```

```
## How would you change the amount of content per newsletter? (e.g. have 2 library updates instead of
rename_at(3, ~ "amount") |>
## How would you change the frequency of the newsletter?
rename_at(4, ~ "frequency") |>
## How would you feel if some minor content of the newsletter was generated by an AI if the maintaine
rename_at(5, ~ "ai") |>
## The current tone of voice in the newsletter is...
rename_at(6, ~ "tone") |>
## How often have you contributed to the newsletter before?
rename_at(7, ~ "contributions") |>
## On a scale of 1-5, how easy is the current process for contributing to the newsletter? Leave this
rename_at(8, ~ "ease") |>
## How do you usually get informed about a new edition of the newsletter?
rename_at(9, ~ "informed") |>
## What about the newsletter do you like and wish to not be changed?
rename_at(10, ~ "like") |>
## What is the most important thing to improve about the newsletter?
rename_at(11, ~ "improve") |>
## Is there something else you would like to tell us?
rename_at(12, ~ "comment") |>
# Rename factors
mutate(
  amount = fct_recode(amount,
    less = "less content; keep only the most important news",
    leave = "leave it as-is",
    more = "more content; add sections for minor news",
    dont_care = "I don't care",
  ),
  frequency = fct_recode(frequency,
    weekly = "make it weekly",
    monthly = "leave it monthly",
    quarterly = "make it quarterly",
    dont_care = "I don't care",
  ),
  ai = fct_recode(ai,
    not_okay = "not okay at all",
    okay = "not ideal, but okay",
    good = "sounds like a good idea",
    love = "I love this",
    dont_care = "I don't care",
  ),
  tone = fct_recode(tone,
    too_formal = "too formal",
    just_right = "just right",
    too_casual = "too informal",
  ),
  contributions = fct_recode(contributions,
    "0" = "never",
    "1" = "once",
    "2-5" = "2-5 times",
    ">5" = "more than 5 times",
  ),
) |>
```

```
  mutate(
    informed = informed |>
      tolower() |>
      str_replace_all("this-week-in-rust", "informed_twir") |>
      str_replace_all("this week in rust", "informed_twir") |>
      str_replace_all("twitter / x", "informed_twitter") |>
      str_replace_all("mastodon", "informed_mastodon") |>
      str_replace_all("rss", "informed_rss") |>
      str_replace_all("reddit", "informed_reddit") |>
      str_replace_all("email subscription to some channel", "informed_email") |>
      str_replace_all("the official rust gamedev discord server", "informed_discord_main") |>
      str_replace_all("some other discord server", "informed_discord_other") |>
      str_replace_all("hacker news", "informed_hacker_news") |>
      str_replace_all("i visit the website every few week", "informed_website")
  )

# The following splits the "informed" column,
# which contains factors concatenated by ;,
# into multiple boolean columns
# Source: <https://stackoverflow.com/a/64412498/5903309>
informed <- unique(unlist(strsplit(dat$informed, ';'), recursive = FALSE))
informed <- informed[!is.na(informed)]

for (level in informed) {
  dat$newcol = rep(0)
  dat <- rename(dat, !!level := newcol)
  dat[grep(level, dat$informed), level] = 1
}

# Add options no one picked
dat$informed_lemmy <- 0
dat$informed_friends <- 0


dat <- dat |>
  dplyr::select(-informed) |>
  dplyr::select(-timestamp)

dat |> write_csv("data_cleaned.csv")
```

Load Data

```
dat <- read_csv("data_cleaned.csv") |>
  mutate(
    excitement =
      as_factor(excitement) |>
      fct_relevel("1", "2", "3", "4", "5"),
    amount = as_factor(amount),
    frequency = as_factor(frequency),
    ai =
      as_factor(ai) |>
      fct_relevel("not_okay", "okay", "dont_care", "good", "love"),
    tone = as_factor(tone),
```

```
    contributions =
      as_factor(contributions) |>
      fct_relevel("0", "1", "2-5", ">5"),
    ease =
      as_factor(ease) |>
      fct_relevel("1", "2", "3", "4", "5"),
    informed_twir = as_factor(informed_twir),
    informed_twitter = as_factor(informed_twitter),
    informed_rss = as_factor(informed_rss),
    informed_reddit = as_factor(informed_reddit),
    informed_email = as_factor(informed_email),
    informed_discord_main = as_factor(informed_discord_main),
    informed_discord_other = as_factor(informed_discord_other),
    informed_hacker_news = as_factor(informed_hacker_news),
    informed_website = as_factor(informed_website),
    informed_mastodon = as_factor(informed_mastodon),
    informed_lemmy = as_factor(informed_lemmy),
    informed_friends = as_factor(informed_friends),
  )
```

```
## Rows: 52 Columns: 22
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (8): amount, frequency, ai, tone, contributions, like, improve, comment
## dbl (14): excitement, ease, informed_discord_main, informed_discord_other, i...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
dat |> summary()
```

```
##  excitement       amount       frequency          ai             tone
##  1: 2       less     : 9   monthly  :29   not_okay :18   too_formal: 6
##  2: 6       leave    :19   dont_care: 9   okay     :18   just_right:45
##  3:12       more     :13   quarterly:11   dont_care: 6   too_casual: 1
##  4:22       dont_care:11   weekly   : 3   good     : 5
##  5:10                                     love     : 5
##
##  contributions   ease        like             improve
##  0  :32        1   : 3   Length:52          Length:52
##  1  : 2        2   : 5   Class :character   Class :character
##  2-5:14        3   : 6   Mode  :character   Mode  :character
##  >5 : 4        4   : 6
##                5   : 3
##                NA's:29
##    comment          informed_discord_main informed_discord_other
##  Length:52          0:37                  0:44
##  Class :character   1:15                  1: 8
##  Mode  :character
##
##
##
##  informed_reddit informed_rss informed_email informed_mastodon informed_twitter
##  0:41            0:33         0:50           0:47              0:48
##  1:11            1:19         1: 2           1: 5              1: 4
```

```
##
##
##
##
##  informed_twir informed_website informed_hacker_news informed_lemmy
##  0:49          0:51             0:51                 0:52
##  1: 3          1: 1             1: 1
##
##
##
##
##  informed_friends
##  0:52
##
##
##
##
##
```

```
dat |> head()
```

```
## # A tibble: 6 x 22
##   excitement amount    frequency ai      tone  contributions ease  like  improve
##   <fct>      <fct>     <fct>     <fct>   <fct> <fct>         <fct> <chr> <chr>
## 1 4          less      monthly   love    too_~ 2-5           2     <NA>  <NA>
## 2 5          leave     dont_care not_ok~ just~ 0             <NA>  <NA>  Please~
## 3 4          leave     monthly   okay    just~ 0             <NA>  <NA>  <NA>
## 4 4          leave     quarterly okay    just~ 0             <NA>  <NA>  <NA>
## 5 4          more      monthly   not_ok~ just~ 0             <NA>  <NA>  <NA>
## 6 4          dont_care monthly   okay    just~ 0             <NA>  <NA>  <NA>
## # i 13 more variables: comment <chr>, informed_discord_main <fct>,
## #   informed_discord_other <fct>, informed_reddit <fct>, informed_rss <fct>,
## #   informed_email <fct>, informed_mastodon <fct>, informed_twitter <fct>,
## #   informed_twir <fct>, informed_website <fct>, informed_hacker_news <fct>,
## #   informed_lemmy <fct>, informed_friends <fct>
```

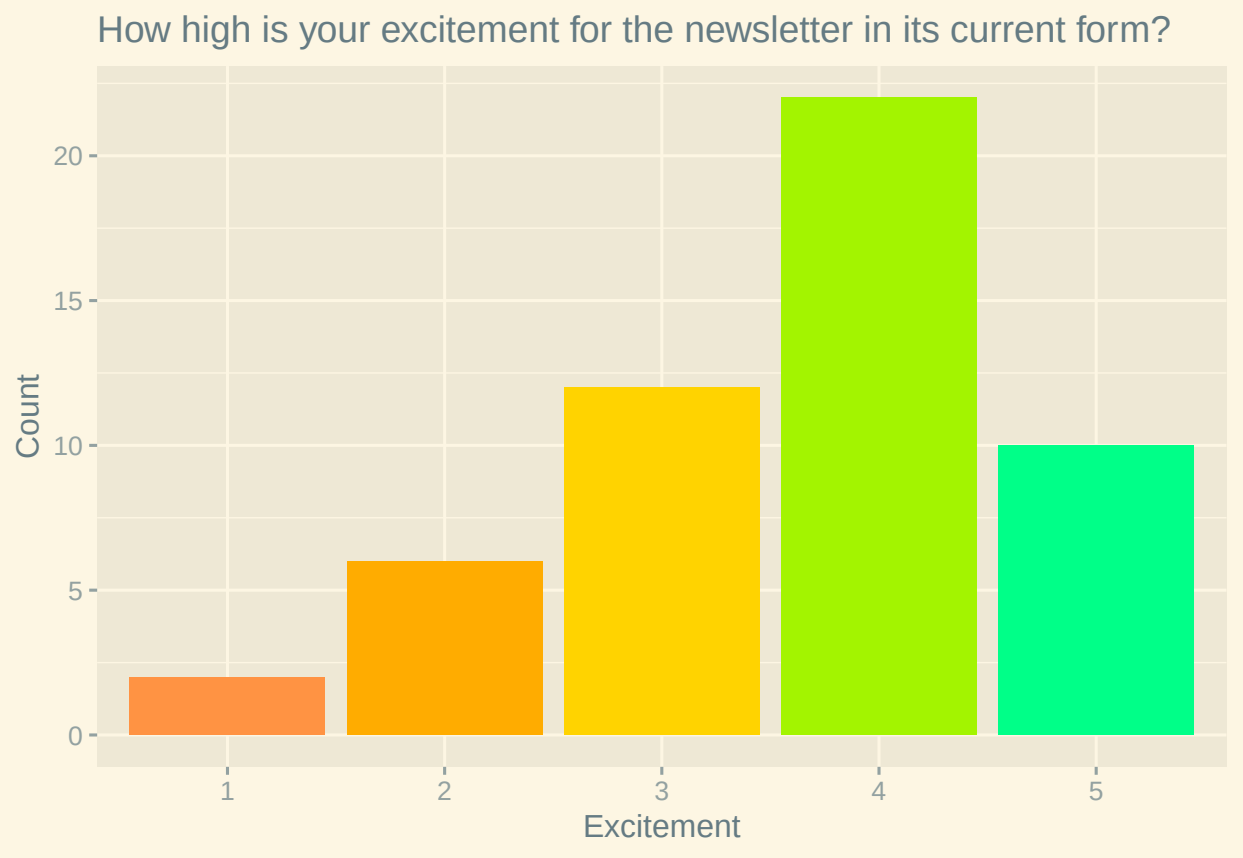
Plots

Excitement

```
scale_values <- c(
  rgb(0, 255, 136, maxColorValue = 255),
  rgb(163, 244, 0, maxColorValue = 255),
  rgb(255, 211, 0, maxColorValue = 255),
  rgb(255, 172, 0, maxColorValue = 255),
  rgb(255, 147, 67, maxColorValue = 255)
)
```

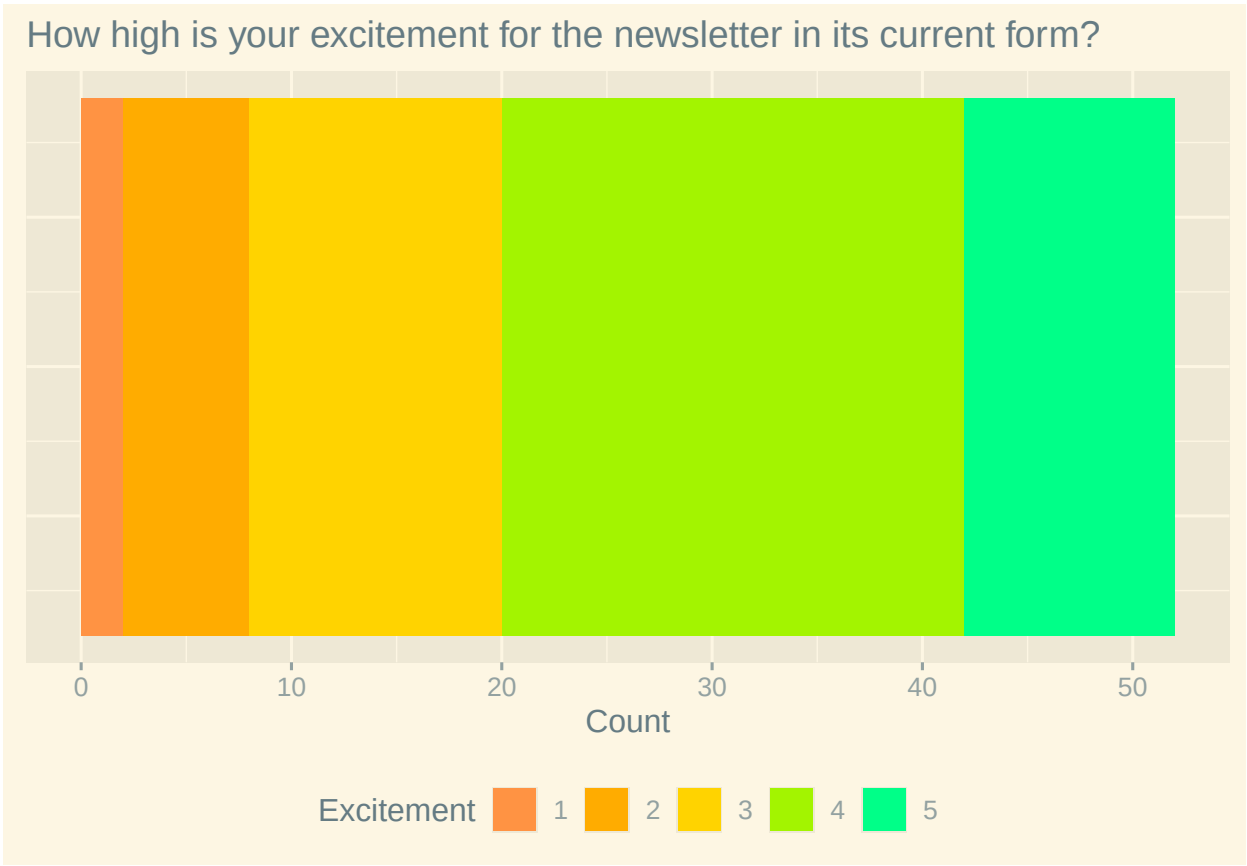
```
dat |>
  count(excitement) |>
  ggplot(aes(x = excitement, y = n, fill = excitement)) +
  scale_fill_manual(values = scale_values |> rev()) +
  geom_col() +
  theme(legend.position = "none") +
  labs(
```

```
    title = "How high is your excitement for the newsletter in its current form?",
    y = "Count",
    x = "Excitement",
    fill = "Excitement"
  )
```




```
dat |>
  count(excitement) |>
  mutate(
    excitement = excitement |>
      fct_reorder(
        as.numeric(excitement),
        .desc = TRUE
      )
  ) |>
  ggplot(aes(x = 1, y = n, fill = excitement)) +
  scale_fill_manual(values = scale_values) +
  geom_col() +
  guides(fill = guide_legend(reverse = TRUE)) +
  theme(legend.position = "bottom") +
  labs(
    title = "How high is your excitement for the newsletter in its current form?",
    y = "Count",
    x = NULL,
    fill = "Excitement"
  ) +
```

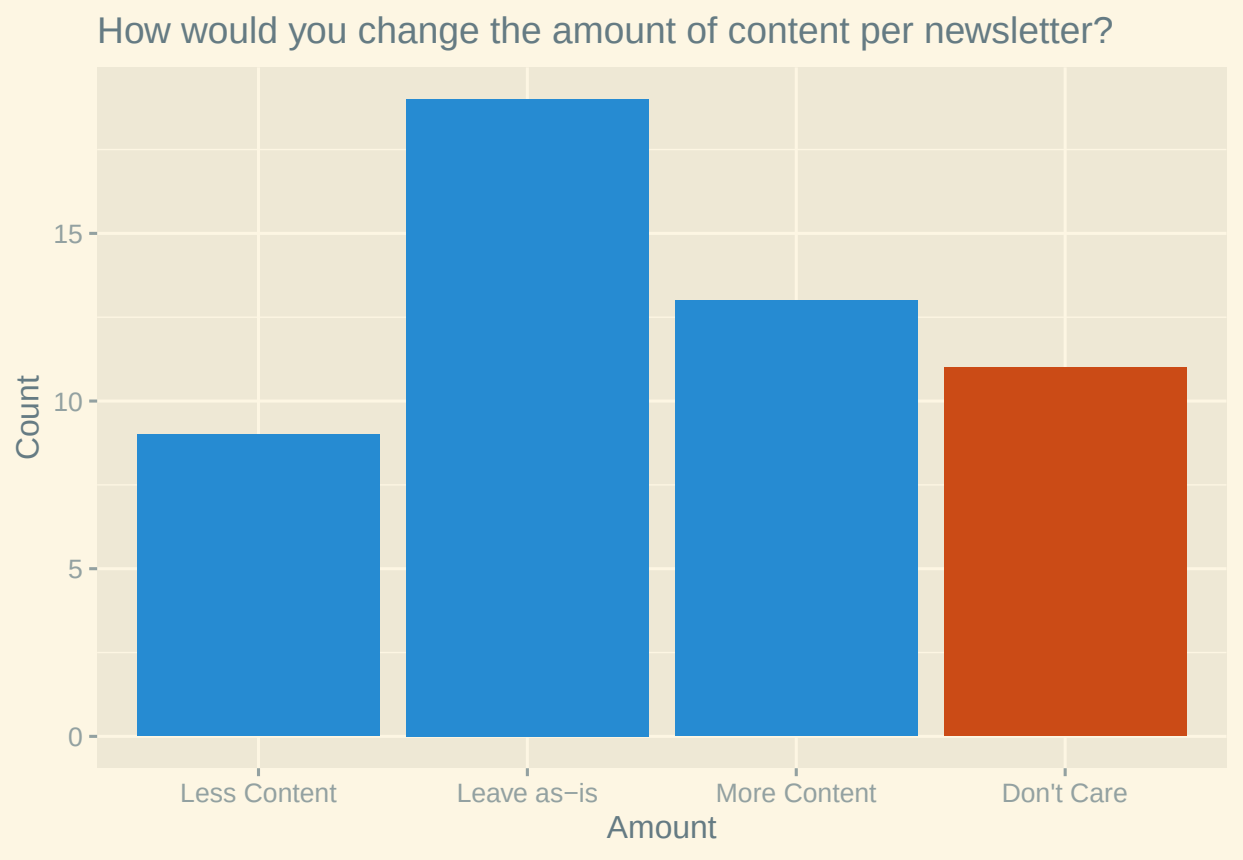
```
  theme(axis.text.y = element_blank(), axis.ticks.y = element_blank()) +
  coord_flip()
```



Amount

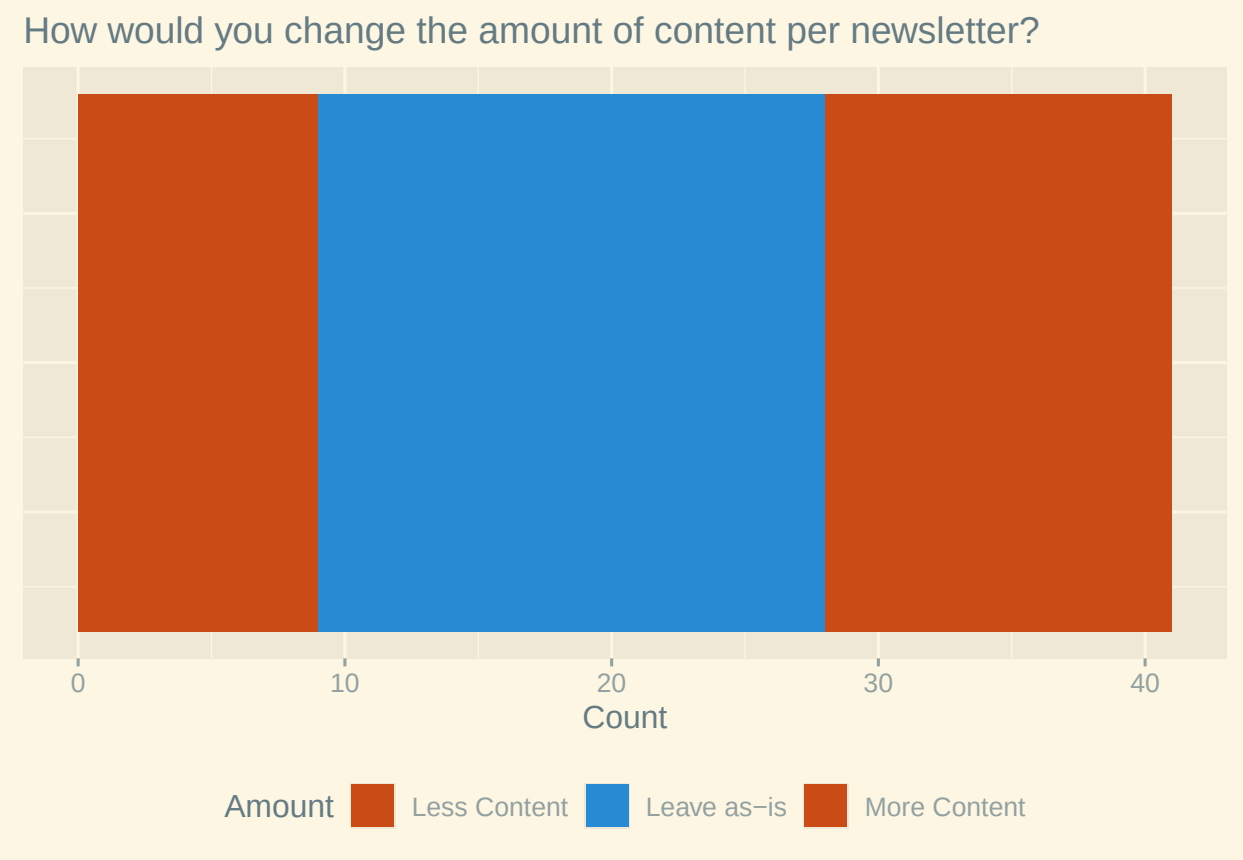
```
dat |>
  mutate(
    amount = amount |>
      fct_relevel("less", "leave", "more", "dont_care") |>
      fct_recode(
        "Less Content" = "less",
        "Leave as-is" = "leave",
        "More Content" = "more",
        "Don't Care" = "dont_care"
      )
  ) |>
  count(amount) |>
  ggplot(aes(x = amount, y = n, fill = amount)) +
  geom_bar(stat = 'identity') +
  theme(legend.position = "none") +
  scale_fill_manual(values = c(palette(1), palette(1), palette(1), palette(2))) +
  labs(
    title = "How would you change the amount of content per newsletter?",
    y = "Count",
    x = "Amount",
```

```
)
```



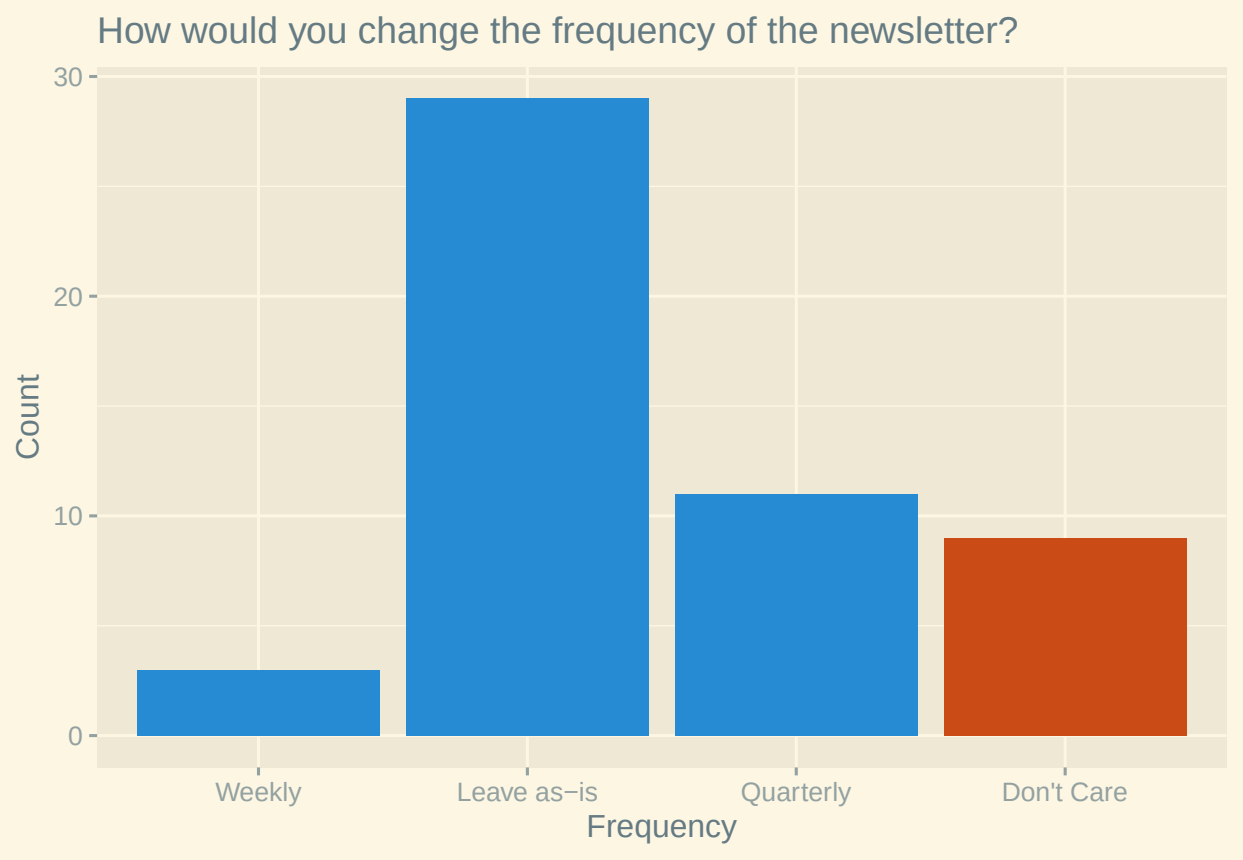

```
dat |>
  filter(amount != "dont_care") |>
  count(amount) |>
  mutate(
    amount = amount |>
      fct_relevel("more", "leave", "less") |>
      fct_recode(
        "Less Content" = "less",
        "Leave as-is" = "leave",
        "More Content" = "more",
      )
  ) |>
  ggplot(aes(x = 1, y = n, fill = amount)) +
  scale_fill_manual(values = c(palette(2), palette(1), palette(2))) +
  geom_bar(stat = 'identity') +
  guides(fill = guide_legend(reverse = TRUE)) +
  theme(legend.position = "bottom") +
  labs(
    title = "How would you change the amount of content per newsletter?",
    y = "Count",
    x = NULL,
    fill = "Amount"
  ) +
  theme(axis.text.y = element_blank(), axis.ticks.y = element_blank()) +
```

```
  coord_flip()
```

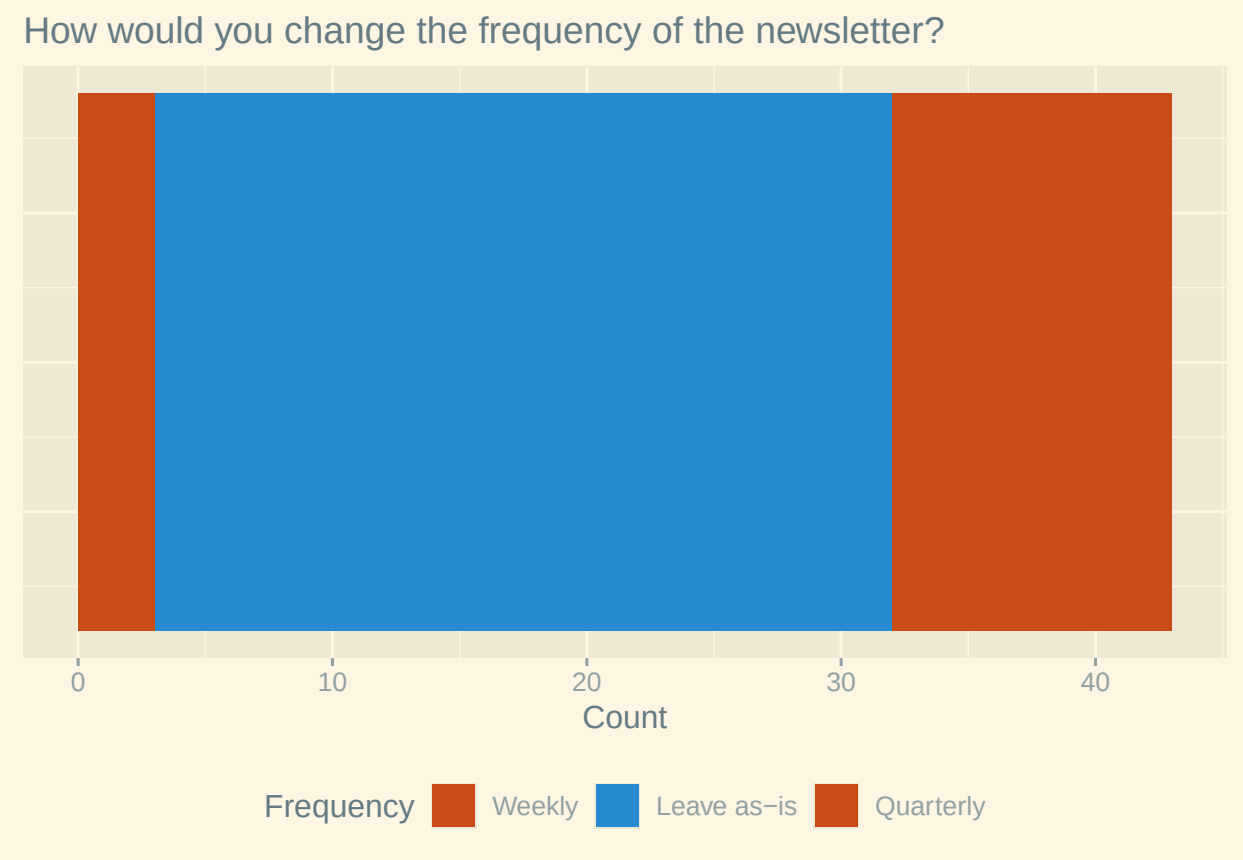


Frequency

```
dat |>
  mutate(
    frequency = frequency |>
      fct_relevel("weekly", "monthly", "quarterly", "dont_care") |>
      fct_recode(
        "Weekly" = "weekly",
        "Leave as-is" = "monthly",
        "Quarterly" = "quarterly",
        "Don't Care" = "dont_care"
      )
  ) |>
  count(frequency) |>
  ggplot(aes(x = frequency, y = n, fill = frequency)) +
  geom_bar(stat = 'identity') +
  theme(legend.position = "none") +
  scale_fill_manual(values = c(palette(1), palette(1), palette(1), palette(2))) +
  labs(
    title = "How would you change the frequency of the newsletter?",
    y = "Count",
    x = "Frequency",
  )
```

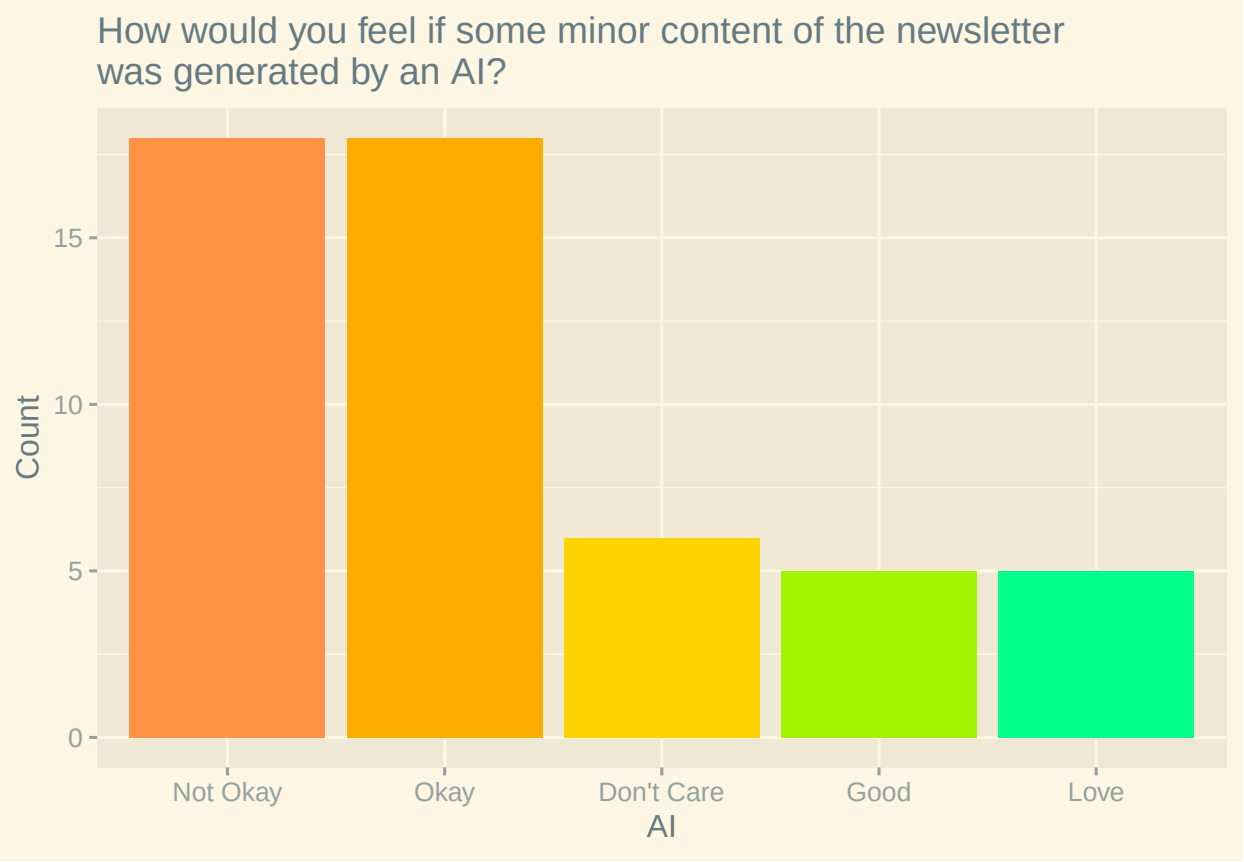



```
dat |>
  filter(frequency != "dont_care") |>
  count(frequency) |>
  mutate(
    frequency = frequency |>
      fct_relevel("quarterly", "monthly", "weekly") |>
      fct_recode(
        "Weekly" = "weekly",
        "Leave as-is" = "monthly",
        "Quarterly" = "quarterly",
      )
  ) |>
  ggplot(aes(x = 1, y = n, fill = frequency)) +
  scale_fill_manual(values = c(palette(2), palette(1), palette(2))) +
  geom_bar(stat = 'identity') +
  guides(fill = guide_legend(reverse = TRUE)) +
  theme(legend.position = "bottom") +
  labs(
    title = "How would you change the frequency of the newsletter?",
    y = "Count",
    x = NULL,
    fill = "Frequency"
  ) +
  theme(axis.text.y = element_blank(), axis.ticks.y = element_blank()) +
  coord_flip()
```

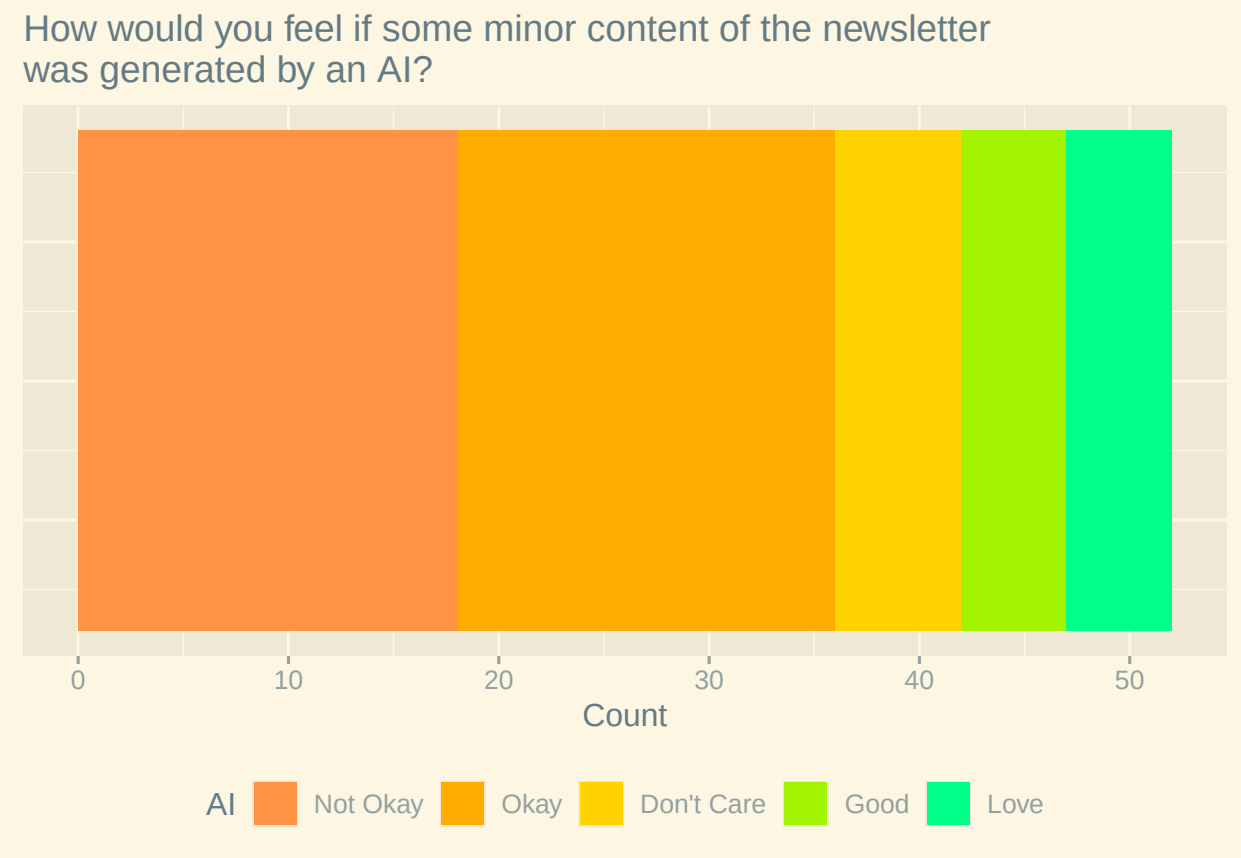


AI

```
dat |>
  mutate(
    ai = ai |>
      fct_relevel("not_okay", "okay", "dont_care", "good", "love") |>
      fct_recode(
        "Not Okay" = "not_okay",
        "Okay" = "okay",
        "Good" = "good",
        "Love" = "love",
        "Don't Care" = "dont_care"
      )
  ) |>
  count(ai) |>
  ggplot(aes(x = ai, y = n, fill = ai)) +
  geom_bar(stat = 'identity') +
  theme(legend.position = "none") +
  scale_fill_manual(values = scale_values |> rev()) +
  labs(
    title = "How would you feel if some minor content of the newsletter\nwas generated by an AI?",
    y = "Count",
    x = "AI",
  )
```

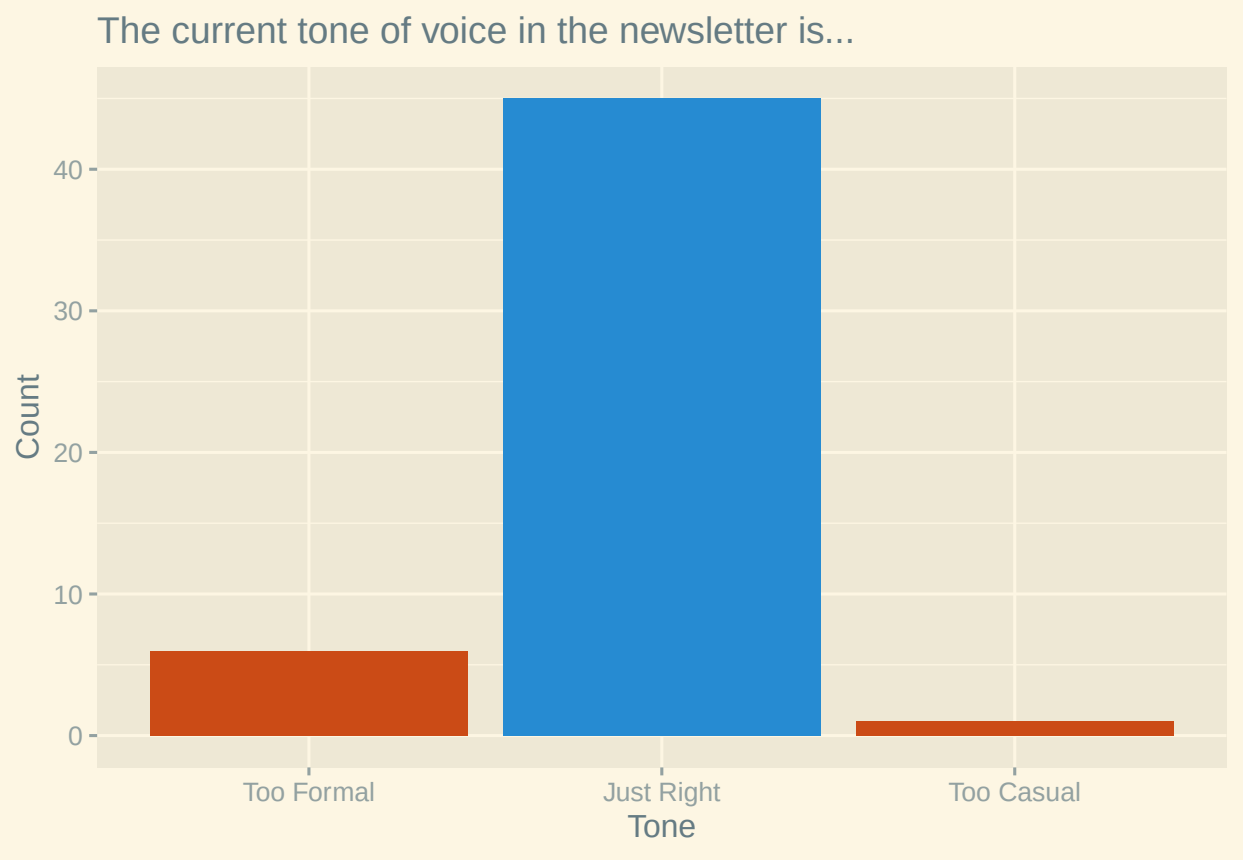



```
dat |>
  count(ai) |>
  mutate(
    ai = ai |>
      fct_relevel("love", "good", "dont_care", "okay", "not_okay") |>
      fct_recode(
        "Not Okay" = "not_okay",
        "Okay" = "okay",
        "Good" = "good",
        "Love" = "love",
        "Don't Care" = "dont_care"
      )
  ) |>
  ggplot(aes(x = 1, y = n, fill = ai)) +
  scale_fill_manual(values = scale_values) +
  geom_bar(stat = 'identity') +
  guides(fill = guide_legend(reverse = TRUE)) +
  theme(legend.position = "bottom") +
  labs(
    title = "How would you feel if some minor content of the newsletter\nwas generated by an AI?",
    y = "Count",
    x = NULL,
    fill = "AI"
  ) +
  theme(axis.text.y = element_blank(), axis.ticks.y = element_blank()) +
  coord_flip()
```

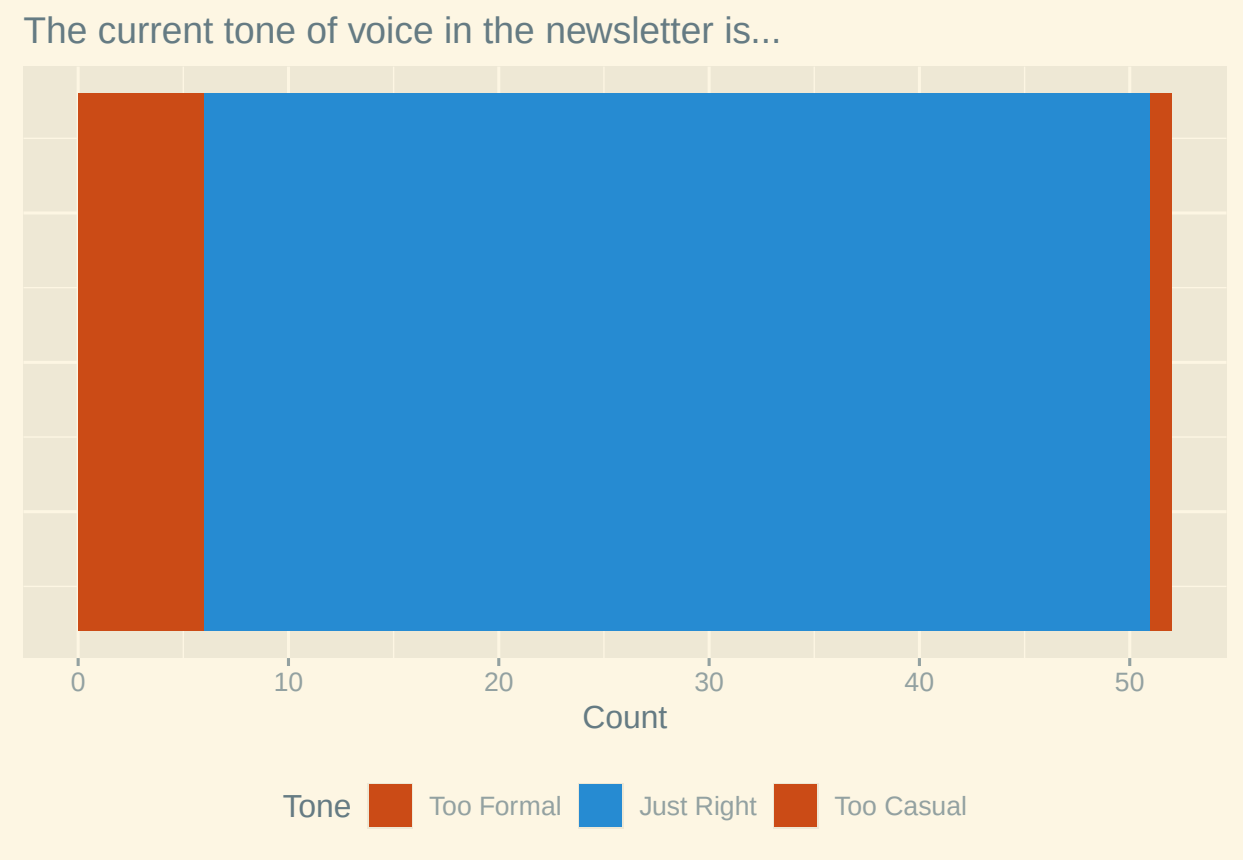


Tone

```
dat |>
  mutate(
    tone = tone |>
      fct_relevel("too_formal", "just_right", "too_casual") |>
      fct_recode(
        "Too Formal" = "too_formal",
        "Just Right" = "just_right",
        "Too Casual" = "too_casual",
      )
  ) |>
  count(tone) |>
  ggplot(aes(x = tone, y = n, fill = tone)) +
  geom_bar(stat = 'identity') +
  theme(legend.position = "none") +
  scale_fill_manual(values = c(palette(2), palette(1), palette(2)))+
    labs(
    title = "The current tone of voice in the newsletter is...",
    y = "Count",
    x = "Tone",
  )
```

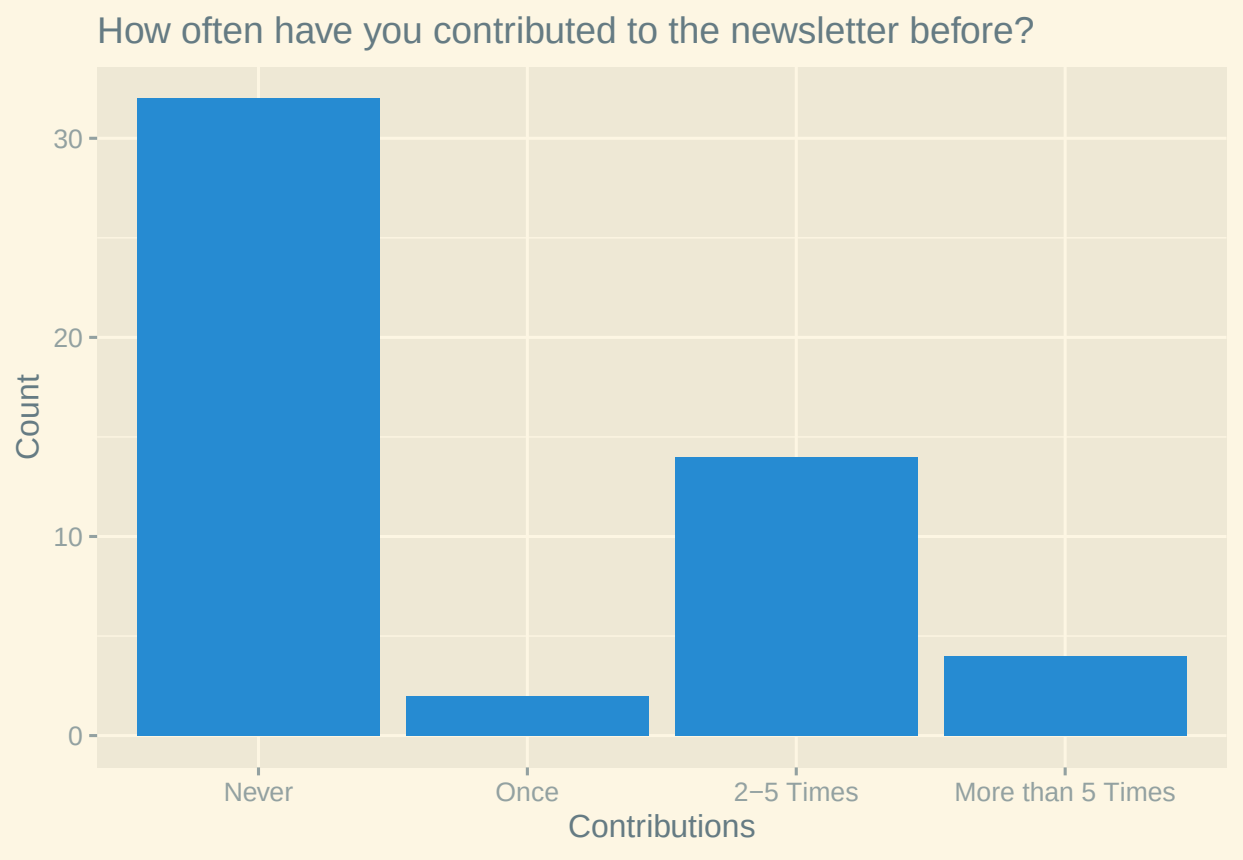



```
dat |>
  count(tone) |>
  mutate(
    tone = tone |>
      fct_relevel("too_casual", "just_right", "too_formal") |>
      fct_recode(
        "Too Formal" = "too_formal",
        "Just Right" = "just_right",
        "Too Casual" = "too_casual",
      )
  ) |>
  ggplot(aes(x = 1, y = n, fill = tone)) +
  scale_fill_manual(values = c(palette(2), palette(1), palette(2))) +
  geom_bar(stat = 'identity') +
  guides(fill = guide_legend(reverse = TRUE)) +
  theme(legend.position = "bottom") +
  labs(
    title = "The current tone of voice in the newsletter is...",
    y = "Count",
    x = NULL,
    fill = "Tone"
  ) +
  theme(axis.text.y = element_blank(), axis.ticks.y = element_blank()) +
  coord_flip()
```



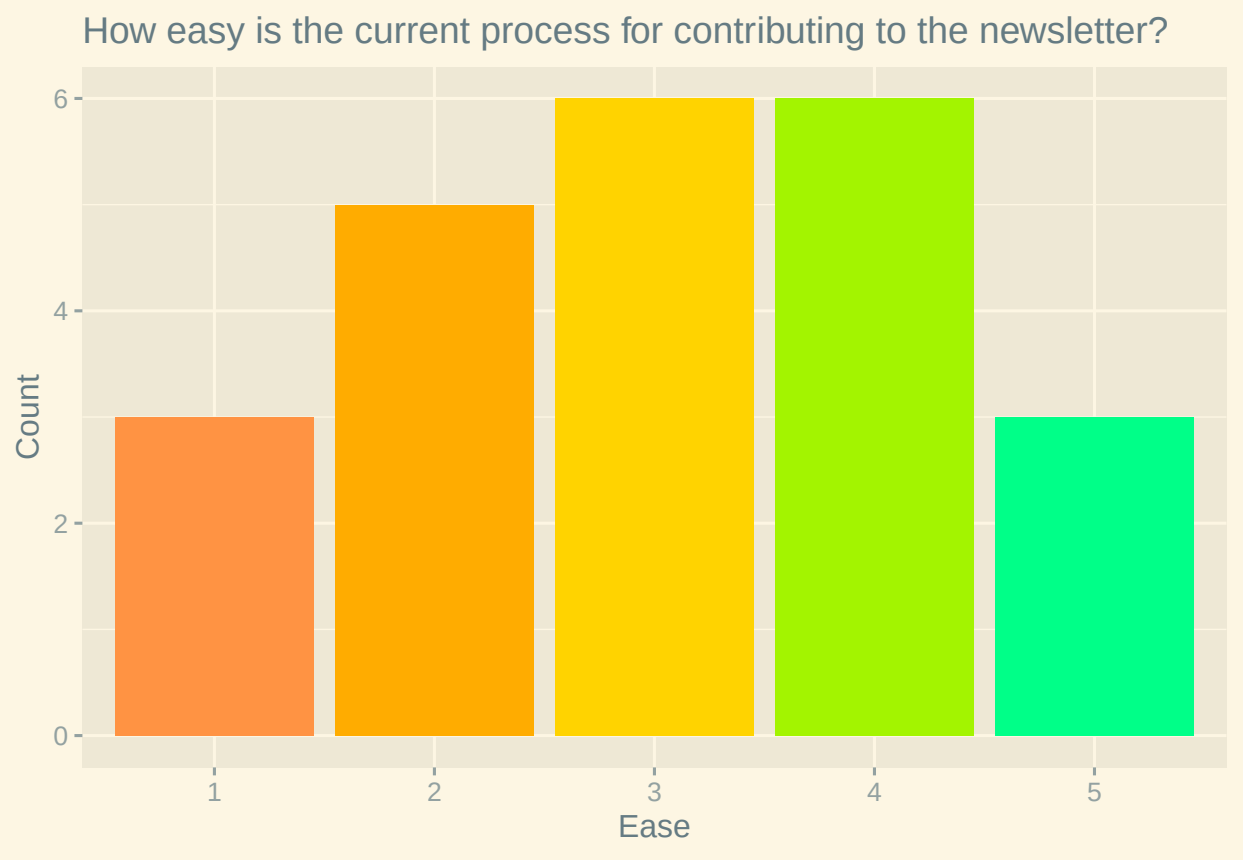
Contributions

```
dat |>
  count(contributions) |>
  mutate(
    contributions = contributions |>
      fct_relevel("0", "1", "2-5", ">5") |>
      fct_recode(
        "Never" = "0",
        "Once" = "1",
        "2-5 Times" = "2-5",
        "More than 5 Times" = ">5",
      )
  ) |>
  ggplot(aes(x = contributions, y = n, fill = contributions)) +
  geom_bar(stat = 'identity') +
  theme(legend.position = "none") +
  scale_fill_manual(values = c(palette(1), palette(1), palette(1), palette(1))) +
  labs(
    title = "How often have you contributed to the newsletter before?",
    y = "Count",
    x = "Contributions",
  )
```

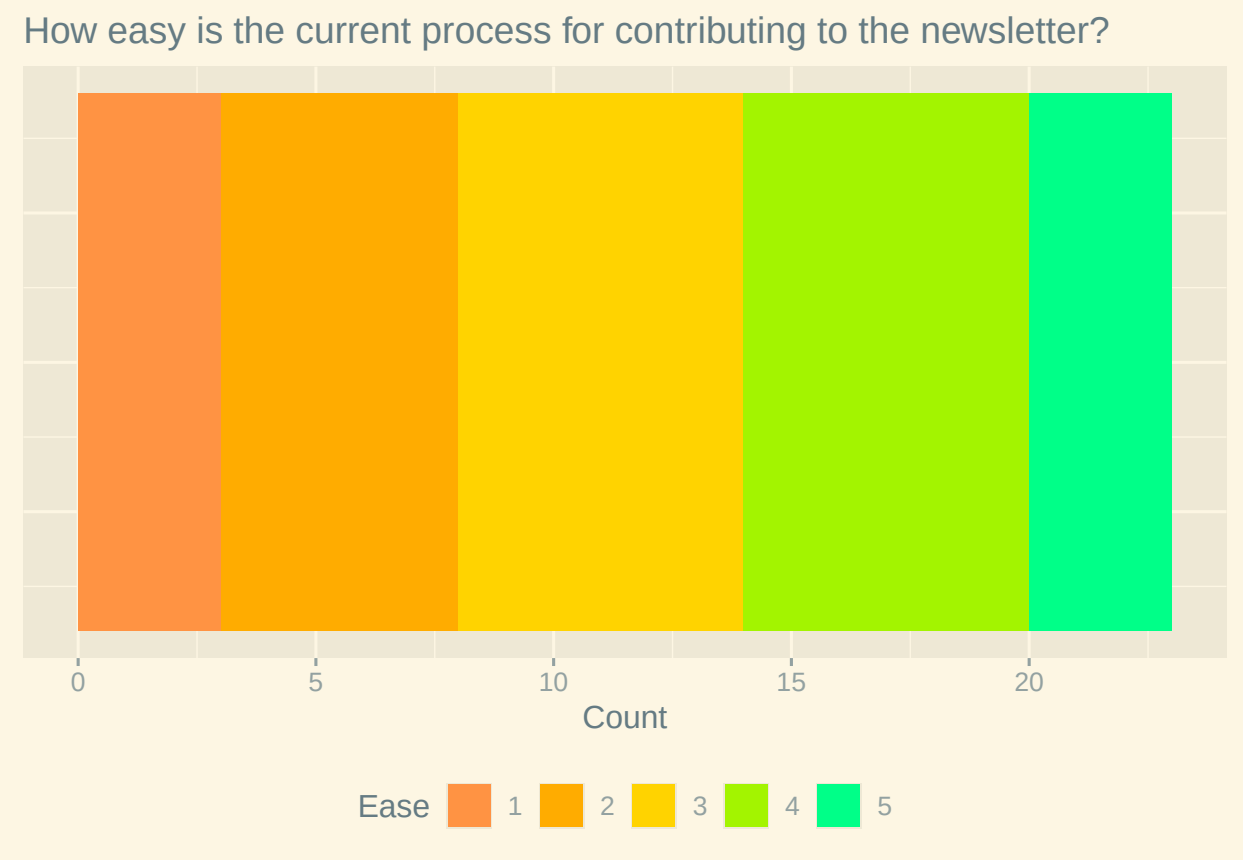


Ease

```
dat |>
  filter(!is.na(ease)) |>
  count(ease) |>
  ggplot(aes(x = ease, y = n, fill = ease)) +
  geom_bar(stat = 'identity') +
  theme(legend.position = "none") +
  scale_fill_manual(values = scale_values |> rev()) +
  labs(
    title = "How easy is the current process for contributing to the newsletter?",
    y = "Count",
    x = "Ease",
  )
```



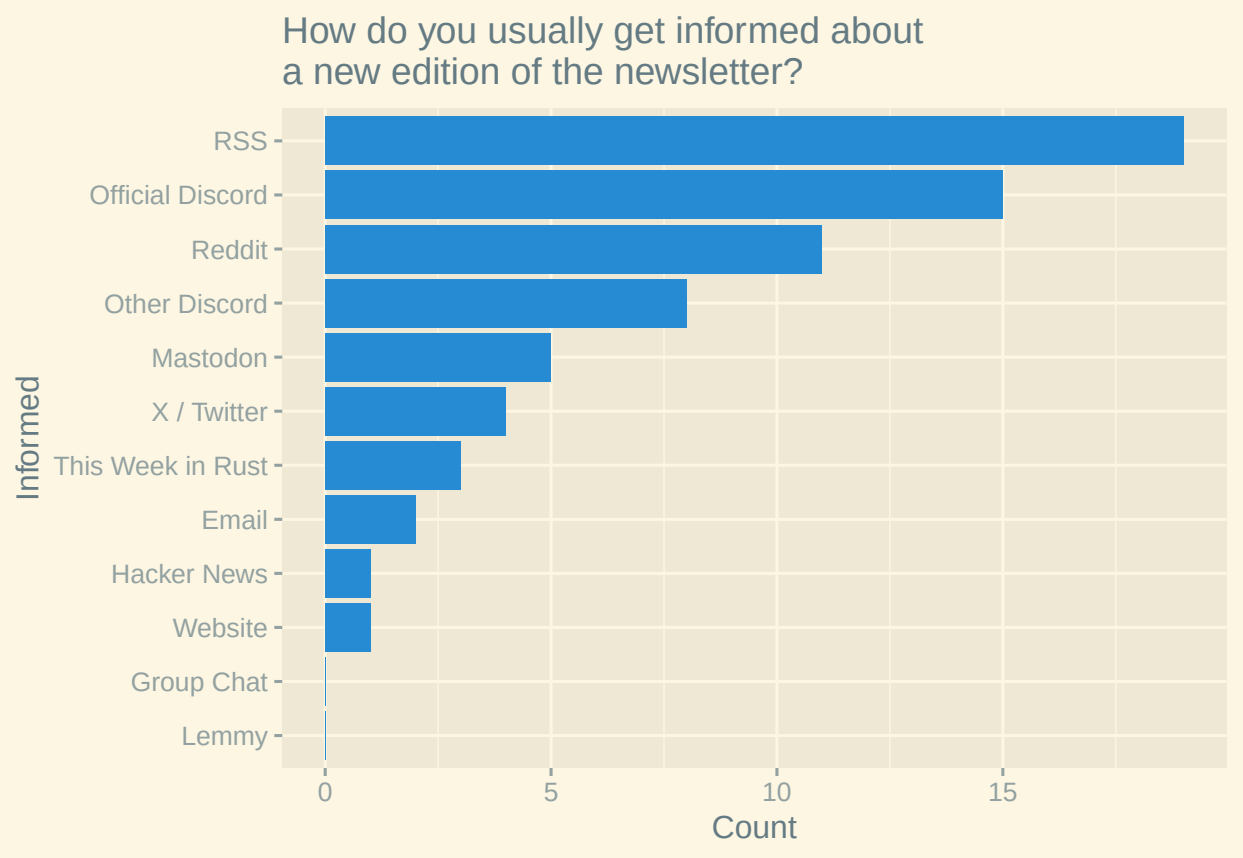

```
dat |>
  filter(!is.na(ease)) |>
  count(ease) |>
  mutate(
    ease = ease |>
      fct_reorder(
        as.numeric(ease),
        .desc = TRUE
      )
  ) |>
  ggplot(aes(x = 1, y = n, fill = ease)) +
  scale_fill_manual(values = scale_values) +
  geom_bar(stat = 'identity') +
  guides(fill = guide_legend(reverse = TRUE)) +
  theme(legend.position = "bottom") +
  labs(
    title = "How easy is the current process for contributing to the newsletter?",
    y = "Count",
    x = NULL,
    fill = "Ease"
  ) +
  theme(axis.text.y = element_blank(), axis.ticks.y = element_blank()) +
  coord_flip()
```



Informed

```
dat |>
  dplyr::select(starts_with("informed")) |>
  pivot_longer(cols = everything()) |>
  filter(value == 1) |>
  count(name) |>
  rbind(
    tibble(name = "informed_friends", n = 0),
    tibble(name = "informed_lemmy", n = 0)
  ) |>
  mutate(
    name = name |>
      fct_recode(
        "X / Twitter" = "informed_twitter",
        "RSS" = "informed_rss",
        "Reddit" = "informed_reddit",
        "Email" = "informed_email",
        "Official Discord" = "informed_discord_main",
        "Other Discord" = "informed_discord_other",
        "Hacker News" = "informed_hacker_news",
        "Website" = "informed_website",
        "This Week in Rust" = "informed_twir",
        "Group Chat" = "informed_friends",
        "Lemmy" = "informed_lemmy",
```

```
      "Mastodon" = "informed_mastodon",
    ) |>
    fct_relevel(c(
      "RSS",
      "Official Discord",
      "Reddit",
      "Other Discord",
      "Mastodon",
      "X / Twitter",
      "This Week in Rust",
      "Email",
      "Hacker News",
      "Website",
      "Group Chat",
      "Lemmy") |> rev())
) |>
ggplot(aes(x = name, y = n)) +
geom_bar(stat = 'identity', fill = palette(1)) +
theme(legend.position = "none") +
labs(
  title = "How do you usually get informed about\na new edition of the newsletter?",
  y = "Count",
  x = "Informed",
) +
coord_flip()
```



Comments

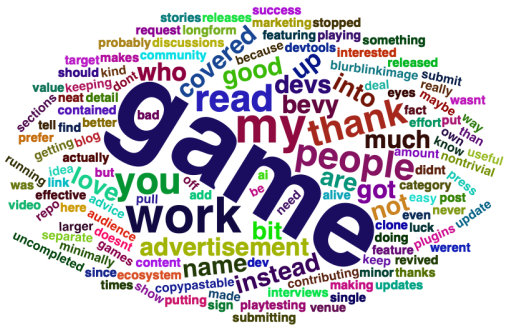
```
clean_words <- function(text) {
  text |>
    str_to_lower() |>
    str_remove_all("[[:punct:]]") |>
    str_remove_all("\\d+") |>
    str_replace_all("\\s+", " ") |>
    str_split(" ") |>
    unlist() |>
    table() |>
    as.data.frame() |>
    arrange(desc(Freq)) |>
    rename(word = Var1, freq = Freq) |>
    mutate(word = as.character(word)) |>
    # common words
    filter(!word %in% c("i", "a", "the", "to", "and", "of", "in", "for", "is", "it", "on", "that", "wit
    # boring results
    filter(!word %in% c("newsletter", "rust"))
}

plot_words <- function(text, filename) {
  graph <- text |>
    clean_words() |>
    wordcloud2()
  # See <https://stackoverflow.com/a/47850987/5903309>
  saveWidget(graph, "tmp.html", selfcontained = F)
  webshot("tmp.html", filename, delay = 5, vwidth = 2000, vheight = 2000)
}
```

```
dat$like |> plot_words("like.png")
```


```
dat$improve |> plot_words("improve.png")
```


```
dat$comment |> plot_words("comment.png")
```



Analysis

Excitement

```
dat_ <- dat$excitement |> as.numeric()
mean_ <- dat_ |> mean()
sd_ <- dat_ |> sd()
ci_ <- t.test(dat_)$conf.int
median_ <- dat_ |> median()
"mean: {mean_}" |> glue()
```

```
## mean: 3.61538461538462
```

```
"standard deviation: {sd_}" |> glue()
```

```
## standard deviation: 1.05074849426018
```

```
"95% confidence interval: [{ci_[1]}, {ci_[2]}]" |> glue()
```

```
## 95% confidence interval: [3.32285436652703, 3.9079148642422]
```

```
"median: {median_}" |> glue()
```

```
## median: 4
```

```
t.test(dat_ - 3)
```

```
##
##  One Sample t-test
##
## data:  dat_ - 3
## t = 4.2233, df = 51, p-value = 9.932e-05
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  0.3228544 0.9079149
## sample estimates:
## mean of x
## 0.6153846
```

Frequency

```
# percentages for each factor
dat |>
  count(frequency) |>
  mutate(
    percentage = n / sum(n) * 100
  )
```

```
## # A tibble: 4 x 3
##   frequency     n percentage
##   <fct>     <int>      <dbl>
## 1 monthly      29      55.8
## 2 dont_care     9      17.3
## 3 quarterly    11      21.2
## 4 weekly        3       5.77
```

```
dat_ <- dat |>
  filter(frequency != "dont_care") |>
  pull(frequency) |>
  as.numeric()
t.test(dat_ - 1.5)
```

```
##
##  One Sample t-test
##
## data:  dat_ - 1.5
## t = 1.3461, df = 42, p-value = 0.1855
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -0.1102973  0.5521578
```

```
## sample estimates:
## mean of x
## 0.2209302
```

Ease

```
dat_ <- dat$ease[!is.na(dat$ease)] |> as.numeric()
mean_ <- dat_ |> mean()
sd_ <- dat_ |> sd()
ci_ <- t.test(dat_)$conf.int
median_ <- dat_ |> median()
"mean: {mean_}" |> glue()
```

```
## mean: 3.04347826086957
```

```
"standard deviation: {sd_}" |> glue()
```

```
## standard deviation: 1.2605287804426
```

```
"95% confidence interval: [{ci_[1]}, {ci_[2]}]" |> glue()
```

```
## 95% confidence interval: [2.49838474030484, 3.58857178143429]
```

```
"median: {median_}" |> glue()
```

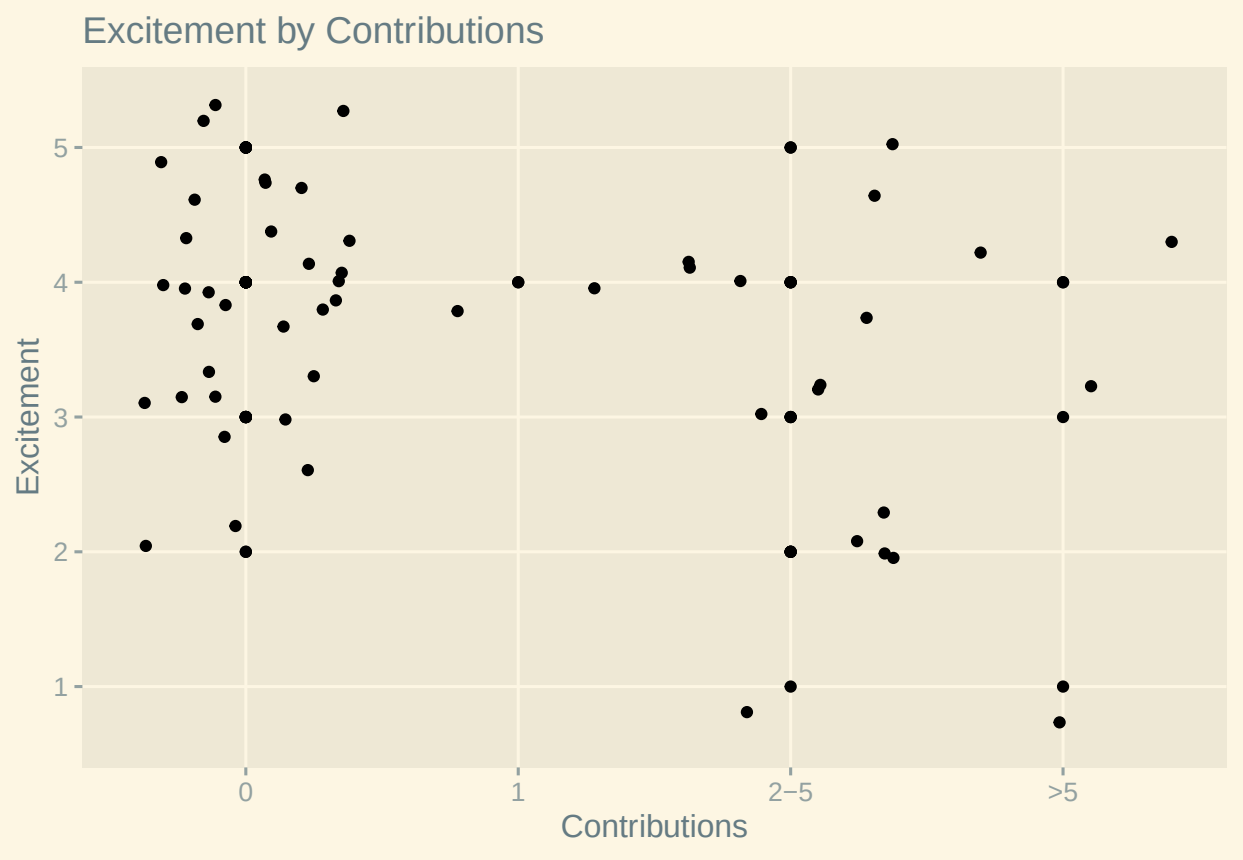
```
## median: 3
```

```
t.test(dat_ - 3)
```

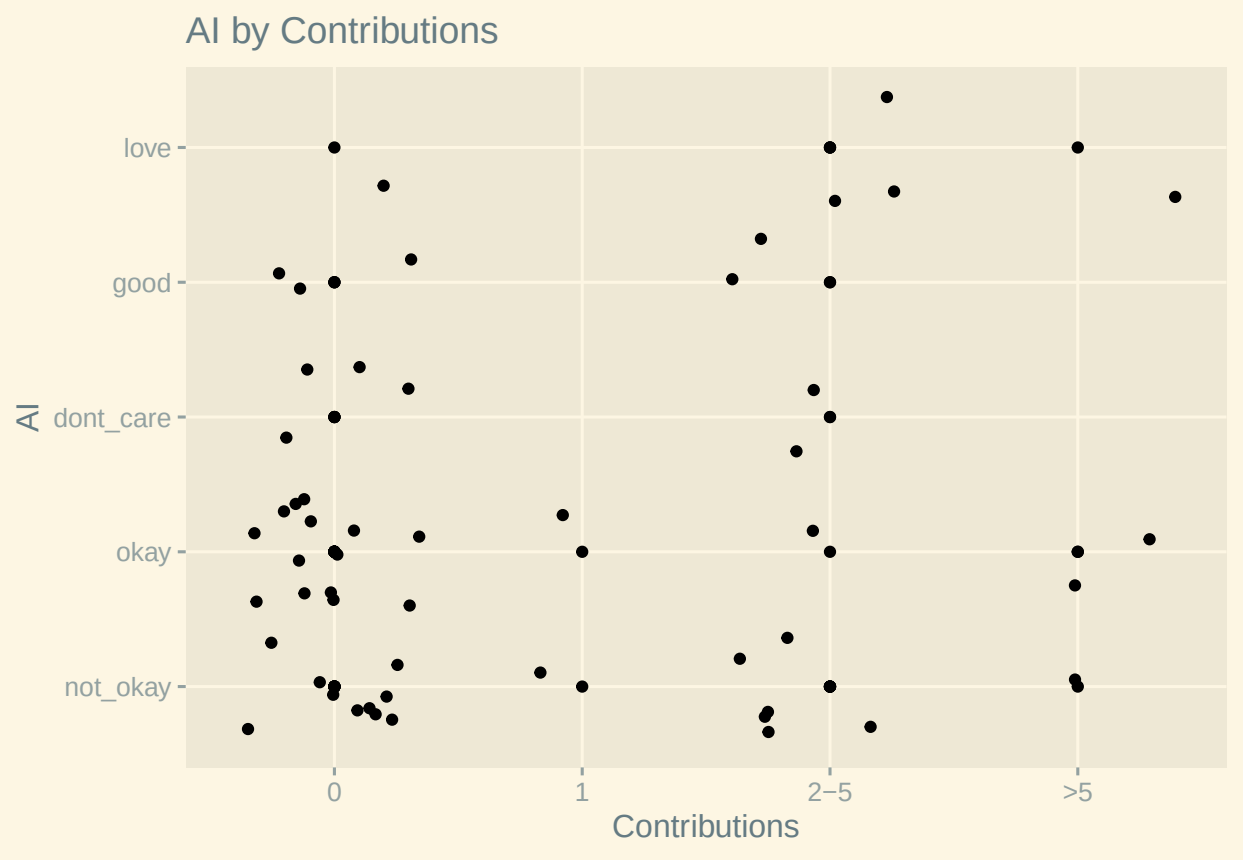
```
##
##  One Sample t-test
##
## data:  dat_ - 3
## t = 0.16542, df = 22, p-value = 0.8701
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -0.5016153  0.5885718
## sample estimates:
##  mean of x
## 0.04347826
```

Correlation

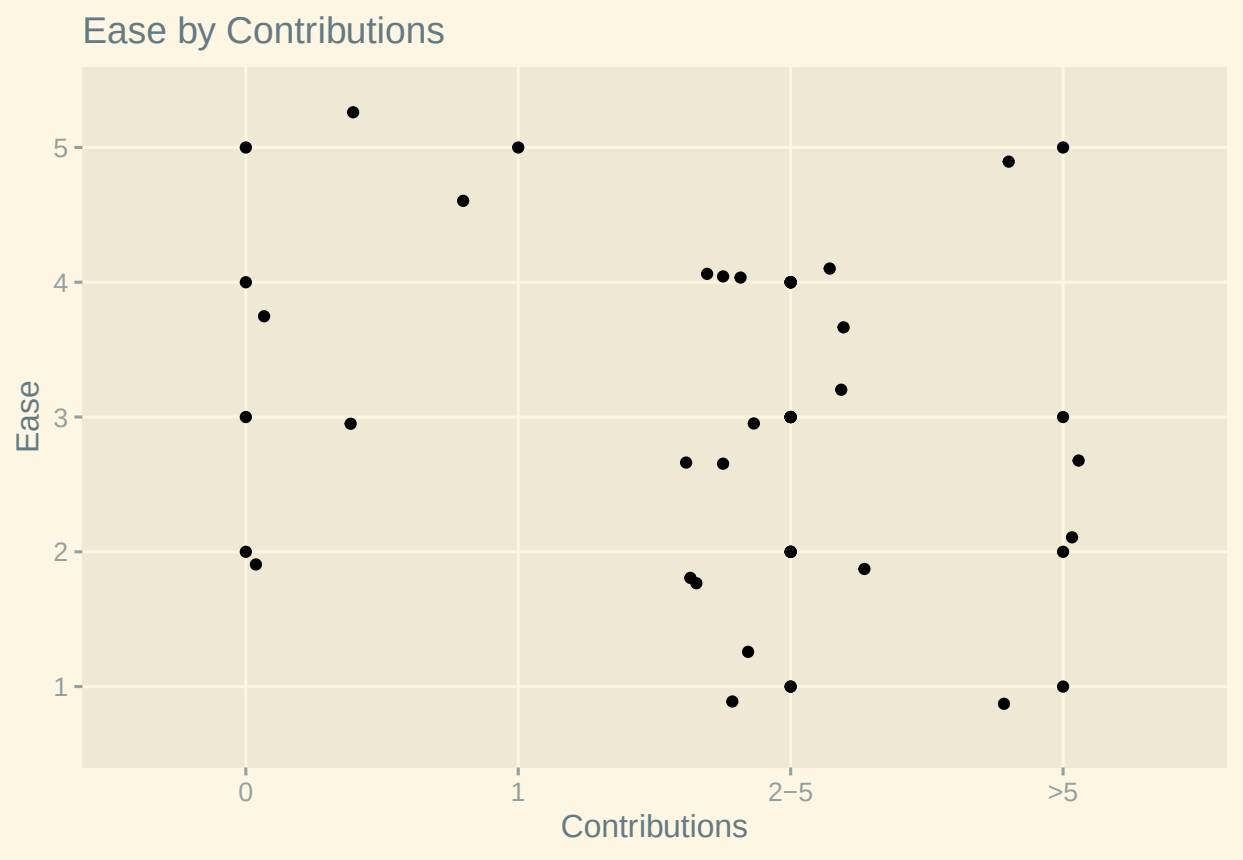
```
# plot excitement by contributions
dat |>
  ggplot(aes(x = contributions, y = excitement)) +
  geom_point() +
  geom_jitter() +
  labs(
    title = "Excitement by Contributions",
    x = "Contributions",
    y = "Excitement"
  )
```




```
# plot ai by contributions
dat |>
  ggplot(aes(x = contributions, y = ai)) +
  geom_point() +
  geom_jitter() +
  labs(
    title = "AI by Contributions",
    x = "Contributions",
    y = "AI"
  )
```




```
# plot ease by contributions
dat |>
  filter(!is.na(ease)) |>
  ggplot(aes(x = contributions, y = ease)) +
  geom_point() +
  geom_jitter() +
  labs(
    title = "Ease by Contributions",
    x = "Contributions",
    y = "Ease"
  )
```




```
JonckheereTerpstraTest(
  excitement ~ contributions,
  dat,
  alternative = "decreasing",
  nperm = 1000,
  exact = FALSE
)
```

```
##
##  Jonckheere-Terpstra test
##
## data:  excitement by contributions
## JT = 257, p-value = 0.012
## alternative hypothesis: decreasing
```

```
JonckheereTerpstraTest(
  ai ~ contributions,
  dat,
  alternative = "increasing",
  nperm = 1000,
  exact = FALSE
)
```

```
##
##  Jonckheere-Terpstra test
##
## data:  ai by contributions
```

```
## JT = 397.5, p-value = 0.287
## alternative hypothesis: increasing
```

```
JonckheereTerpstraTest(
  ease ~ contributions,
  dat |> filter(!is.na(ease)),
  alternative = "decreasing",
  nperm = 1000,
  exact = FALSE
)
```

```
##
##  Jonckheere-Terpstra test
##
## data:  ease by contributions
## JT = 55, p-value = 0.13
## alternative hypothesis: decreasing
```