Informal Rust Gamedev in 2024 Survey

Jan Hohenheim

2024-05-04

Dependencies

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(glue)
library(ggthemes)
library(latex2exp)
library(glmnet)
```

```
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
##
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
##
## Loaded glmnet 4.1-8
```

```
library(MASS)
```

```
##
## Attaching package: 'MASS'
##
## The following object is masked from 'package:dplyr':
##
##     select
```

```
library(rstatix)
```

```
##
## Attaching package: 'rstatix'
##
```

```
## The following object is masked from 'package:MASS':
##
##     select
##
## The following object is masked from 'package:stats':
##
##     filter
```

```
library(ordinal)
```

```
##
## Attaching package: 'ordinal'
##
## The following object is masked from 'package:dplyr':
##
##     slice
```

```
theme_set(theme_solarized_2())
```

Data Cleaning

See https://www.reddit.com/r/rust_gamedev/comments/1cka6n8/informal_rust_gamedev_in_2024_survey_results/

```
dat.raw <- read_csv("data_original.csv")
```

```
## Rows: 410 Columns: 17
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (4): Timestamp, How are you currently using Rust to make games? Select ...
## dbl (13): What are the biggest barriers to your success when making games in...
##
## 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") |>
  ## How are you currently using Rust to make games?
  rename_at(2, ~ "usage") |>
  ## What Rust-based game engine do you primarily use?
  rename_at(3, ~ "engine") |>
  ## Long compile and iteration times
  rename_at(4, ~ "bad_iteration_time") |>
  ## Problems with Rust itself (other than compile times)
  rename_at(5, ~ "bad_rust") |>
  ## Problems in platform-abstracting crates like winit or wgpu
  rename_at(6, ~ "bad_abstraction") |>
  ## Inadequate learning materials or docs
  rename_at(7, ~ "bad_docs") |>
  ## Poor tooling for artists and game designers
  rename_at(8, ~ "bad_tooling") |>
  ## Difficulty paying to get open source problems fixed
  rename_at(9, ~ "bad_paying_for_bugs") |>
  ## Lack of console support
```

```
  rename_at(10, ~ "bad_console") |>
  ## Immature mobile support
  rename_at(11, ~ "bad_mobile") |>
  ## Immature web support
  rename_at(12, ~ "bad_web") |>
  ## Bugs in the engine I use
  rename_at(13, ~ "bad_engine_bugs") |>
  ## Missing features in the engine I use
  rename_at(14, ~ "bad_engine_features") |>
  ## Difficulty hiring experts who know Rust
  rename_at(15, ~ "bad_hiring") |>
  ## Poor performance
  rename_at(16, ~ "bad_performance") |>
  ## If you could magically add or fix three things about Rust itself, what would they be?
  rename_at(17, ~ "magic_fix") |>
  # Rename factors
  mutate(usage = fct_recode(usage,
    gamedev_serious_hobby = "I have at least one serious hobbyist project that I have or am planning to
    tooling_creator = "Actually, I only use Rust to make game engines or tools for gamedev.",
    gamedev_commercial_solo = "I work by myself, but have a project that I have or am planning to sell.
    gamedev_financial_support = "I work by myself or in a tiny team, and am attempting to support mysel
    tooling_company = "I am part of a company that uses Rust game tools to make things that are not game
    gamedev_learner = "I'm still learning.",
  )) |>
  # Classify custom answers. There's probably a better way to do this, sorry
  mutate(usage = usage |>
    fct_recode(gamedev_serious_hobby = "I've used Rust for making tools while working in the games indus
    fct_recode(gamedev_serious_hobby = "I am part of a large open source game written in Rust") |>

    fct_recode(gamedev_commercial_solo = "I have a game engine and game editor in rust that I am looking

    fct_recode(gamedev_financial_support = "Both make games and tools using Rust in a tiny team to suppo
    fct_recode(gamedev_financial_support = "I am part of a company that is using Rust to make games.")

    fct_recode(tooling_company = "I work for Foresight, making CAD tools using bevy and rust.") |>
    fct_recode(tooling_company = "I use Rust game tools for academic research.") |>

    fct_recode(gamedev_learner = "I write rust code but not much game-dev but I dabbled with bevy a litt

    fct_recode(gamedev_casual_hobby = "Hobbyist game developer") |>
    fct_recode(gamedev_casual_hobby = "\"I have at least one serious hobbyist project\" … for which \"sl
    fct_recode(gamedev_casual_hobby = "Only as a hobby") |>
    fct_recode(gamedev_casual_hobby = "I am still in university, so not working commercially, but use Ru
    fct_recode(gamedev_casual_hobby = "I use rust to make games primarily as a hobby, but technically ha
    fct_recode(gamedev_casual_hobby = "I use Bevy to create games and teach my son about developing and
    fct_recode(gamedev_casual_hobby = "I sometimes use Rust/Bevy for Game Jam entries as a change from C
    fct_recode(gamedev_casual_hobby = "I make games for my kids") |>

    fct_recode(gamedev_quit = "I used to make games in Rust.") |>
    fct_recode(gamedev_quit = "Recently moved game project away from Rust") |>
    fct_recode(gamedev_quit = "im not using rust for game dev anymore - turns out it sucks ass for it")
    fct_recode(gamedev_quit = "Tried Rust for gamedev. Too much ceremony needed for everything. It was n
    fct_recode(gamedev_quit = "I was working by myself to financially support with games made in Rust, b
```

```
    fct_recode(gamedev_quit = "I have made games in Rust in the past") |>

    fct_recode(other = "Paid contractor making a metaverse client") |>
    fct_recode(other = "I worked for a games company before") |>
    fct_recode(other = "I work with Rust outside of games but I want to make games in it when the tools
    fct_recode(other = "I was \"I  have at least one serious hobbyist project that I have or am planning
    fct_recode(other = "I use Rust game tools to make things that are not games as a hobby or passion."
    fct_recode(other = "I freelance for companies shipping AAA games.") |>
    fct_recode(other = "I don't make games in Rust.")
) |>
  mutate(engine = factor(engine) |>
    fct_recode(fyrox = "Fyrox") |>
    fct_recode(fyrox = "Been experimenting, but mostly fyrox.") |>
    fct_recode(fyrox = "Tried Fyrox. Great engine but Rust was the limitation.") |>
    fct_recode(bevy = "Bevy") |>
    fct_recode(bevy_extra = "Bevy + In House Engine") |>
    fct_recode(bevy_extra = "both Bevy & no-engine / in-house, depending on client and project") |>
    fct_recode(bevy_extra = "In house and Bevy") |>
    fct_recode(chuot = "Chuột") |>
    fct_recode(comfy = "Comfy") |>
    fct_recode(custom = "Custom-written wgpu-based 2D engine (for Visual Novels)") |>
    fct_recode(custom = "No engine / in-house engine") |>
    fct_recode(custom = "My custom engine") |>
    fct_recode(gamercade = "Gamercade") |>
    fct_recode(godot = "gdnative (Rust and Godot)") |>
    fct_recode(godot = "Godot") |>
    fct_recode(godot = "Godot with rust bindings (gdext)") |>
    fct_recode(godot = "Godot-GDNative") |>
    fct_recode(godot = "godot-rust") |>
    fct_recode(godot = "Godot-Rust") |>
    fct_recode(godot = "Godot + gdnative") |>
    fct_recode(godot = "Godot/gdext") |>
    fct_recode(raylib = "Raylib Rust bindings") |>
    fct_recode(raylib = "Raylib-ffi") |>
    fct_recode(none = "no engine, but we use specs + wgpu + conrod/iced for gui") |>
    fct_recode(none = "No engine with wgpu, bevy ecs, egui.") |>
    fct_recode(none = "No engine") |>
    fct_recode(none = "wgpu") |>
    fct_recode(none = "SDL") |>
    fct_recode(none = "Rend3/WGPU") |>
    fct_recode(none = "None") |>
    fct_recode(tetra = "Tetra") |>
    fct_recode(speedy2d = "Speedy2D") |>
    fct_recode(quad = "Macroquad") |>
    fct_recode(quad = "miniquad") |>
    fct_recode(piston = "Piston") |>
    fct_recode(ggez = "Good Web Game") |>
    fct_recode(other = "I have only tried Fyrox and Bevy but both are currently lacking")
  )

dat |> summary()
```

```
##   timestamp                                usage         engine
##  Length:410         gamedev_serious_hobby    :127   bevy    :289
```

```
##  Class :character   gamedev_learner          : 83   custom : 63
##  Mode  :character   gamedev_commercial_solo  : 51   quad   : 15
##                     gamedev_financial_support: 44   godot  :  9
##                     tooling_creator          : 36   none   :  7
##                     other                    : 30   ggez   :  6
##                     (Other)                  : 39   (Other): 21
##  bad_iteration_time    bad_rust    bad_abstraction    bad_docs
##  Min.   :0.000      Min.   :0.00   Min.   :0.000   Min.   :0.000
##  1st Qu.:1.000      1st Qu.:0.00   1st Qu.:0.000   1st Qu.:1.000
##  Median :2.000      Median :1.00   Median :1.000   Median :2.000
##  Mean   :2.473      Mean   :1.32   Mean   :1.383   Mean   :2.012
##  3rd Qu.:4.000      3rd Qu.:2.00   3rd Qu.:2.000   3rd Qu.:3.000
##  Max.   :5.000      Max.   :5.00   Max.   :5.000   Max.   :5.000
##
##   bad_tooling    bad_paying_for_bugs  bad_console      bad_mobile
##  Min.   :0.000   Min.   :0.0000      Min.   :0.000   Min.   :0.000
##  1st Qu.:1.000   1st Qu.:0.0000      1st Qu.:0.000   1st Qu.:0.000
##  Median :2.000   Median :0.0000      Median :0.000   Median :0.000
##  Mean   :2.307   Mean   :0.5488      Mean   :1.117   Mean   :1.278
##  3rd Qu.:4.000   3rd Qu.:1.0000      3rd Qu.:2.000   3rd Qu.:2.000
##  Max.   :5.000   Max.   :5.0000      Max.   :5.000   Max.   :5.000
##
##     bad_web      bad_engine_bugs  bad_engine_features   bad_hiring
##  Min.   :0.000   Min.   :0.0000   Min.   :0.000       Min.   :0.0000
##  1st Qu.:0.000   1st Qu.:0.0000   1st Qu.:1.000       1st Qu.:0.0000
##  Median :0.000   Median :1.0000   Median :3.000       Median :0.0000
##  Mean   :1.102   Mean   :0.9146   Mean   :2.478       Mean   :0.6512
##  3rd Qu.:2.000   3rd Qu.:1.0000   3rd Qu.:4.000       3rd Qu.:1.0000
##  Max.   :5.000   Max.   :5.0000   Max.   :5.000       Max.   :5.0000
##
##  bad_performance   magic_fix
##  Min.   :0.0000   Length:410
##  1st Qu.:0.0000   Class :character
##  Median :0.0000   Mode  :character
##  Mean   :0.5854
##  3rd Qu.:1.0000
##  Max.   :5.0000
##
```

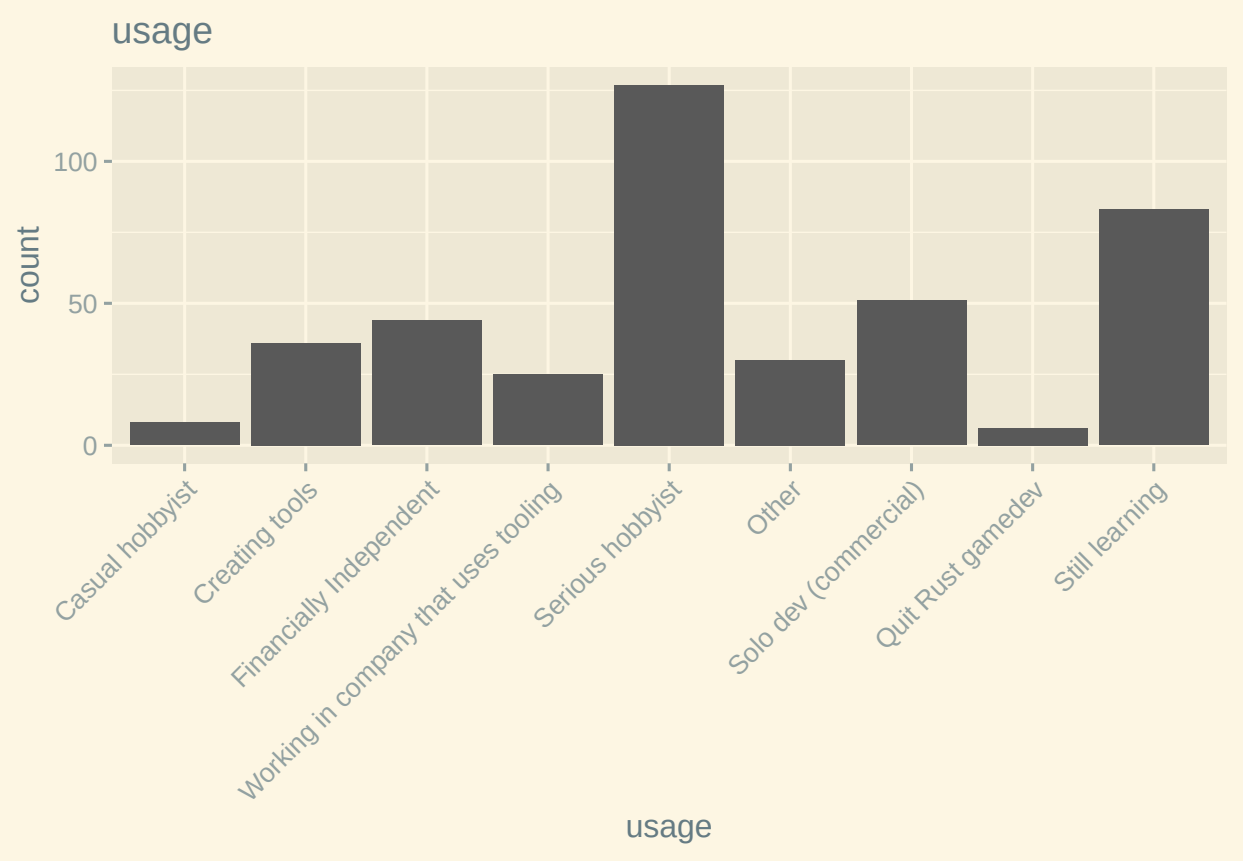
```
dat |> head()
```

```
## # A tibble: 6 x 17
##   timestamp    usage engine bad_iteration_time bad_rust bad_abstraction bad_docs
##   <chr>        <fct> <fct>               <dbl>    <dbl>           <dbl>    <dbl>
## 1 4/29/2024 1~ game~ bevy                    0        2               2        0
## 2 4/29/2024 1~ game~ bevy                    1        0               0        1
## 3 4/29/2024 1~ game~ bevy                    4        2               2        1
## 4 4/29/2024 1~ game~ none                    0        0               1        0
## 5 4/29/2024 1~ game~ bevy                    0        0               0        1
## 6 4/29/2024 1~ game~ bevy                    0        1               3        4
## # i 10 more variables: bad_tooling <dbl>, bad_paying_for_bugs <dbl>,
## #   bad_console <dbl>, bad_mobile <dbl>, bad_web <dbl>, bad_engine_bugs <dbl>,
## #   bad_engine_features <dbl>, bad_hiring <dbl>, bad_performance <dbl>,
## #   magic_fix <chr>
```

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

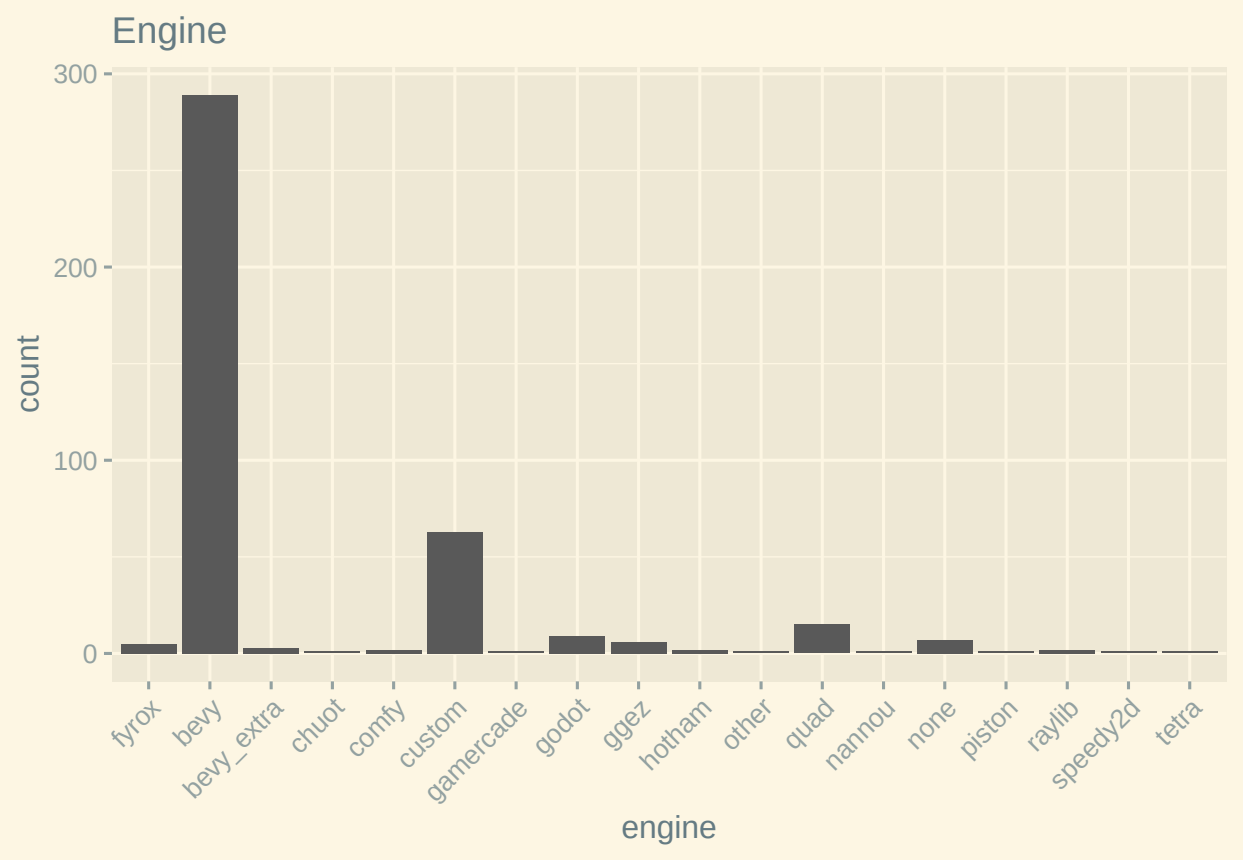
Generic Plots

```
dat |>
  ggplot(aes(x = usage)) +
  geom_bar() +
  ggtitle("usage") +
  # use abbreviated labels
  scale_x_discrete(labels = c(
    "gamedev_serious_hobby" = "Serious hobbyist",
    "tooling_creator" = "Creating tools",
    "gamedev_commercial_solo" = "Solo dev (commercial)",
    "gamedev_financial_support" = "Financially Independent",
    "tooling_company" = "Working in company that uses tooling",
    "gamedev_learner" = "Still learning",
    "gamedev_casual_hobby" = "Casual hobbyist",
    "gamedev_quit" = "Quit Rust gamedev",
    "other" = "Other"
  )) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

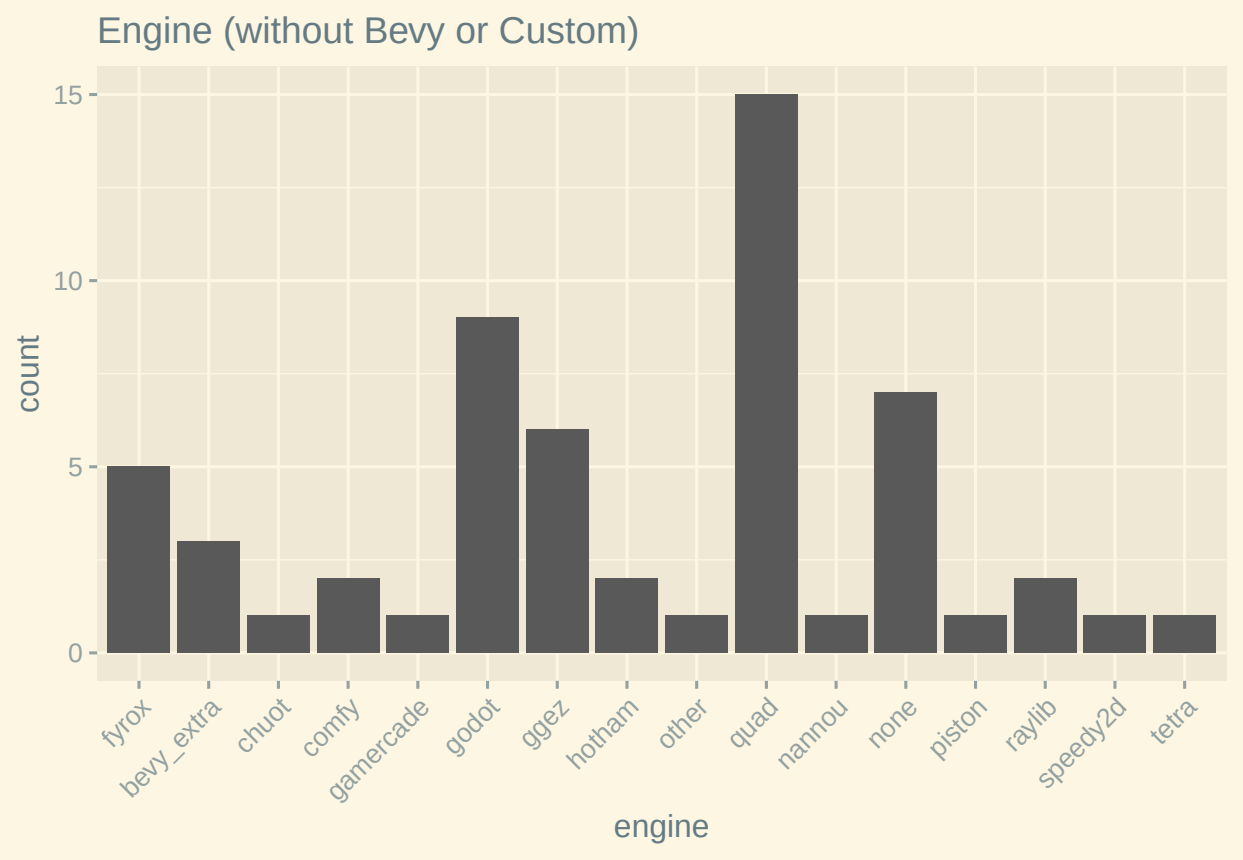



```
dat |>
  ggplot(aes(x = engine)) +
  geom_bar() +
```

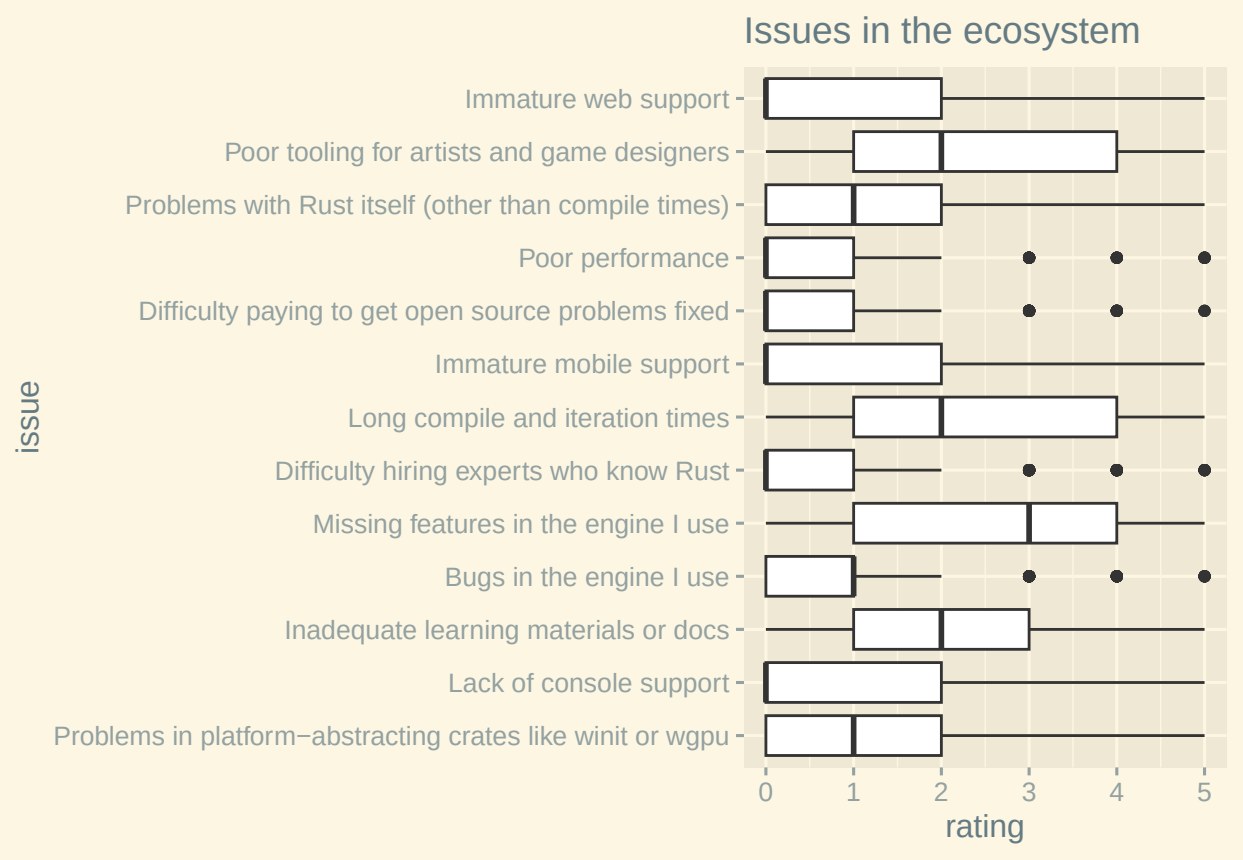
```
  ggtitle("Engine") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```




```
dat |>
  filter(engine != "bevy") |>
  filter(engine != "custom") |>
  ggplot(aes(x = engine)) +
  geom_bar() +
  ggtitle("Engine (without Bevy or Custom)") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



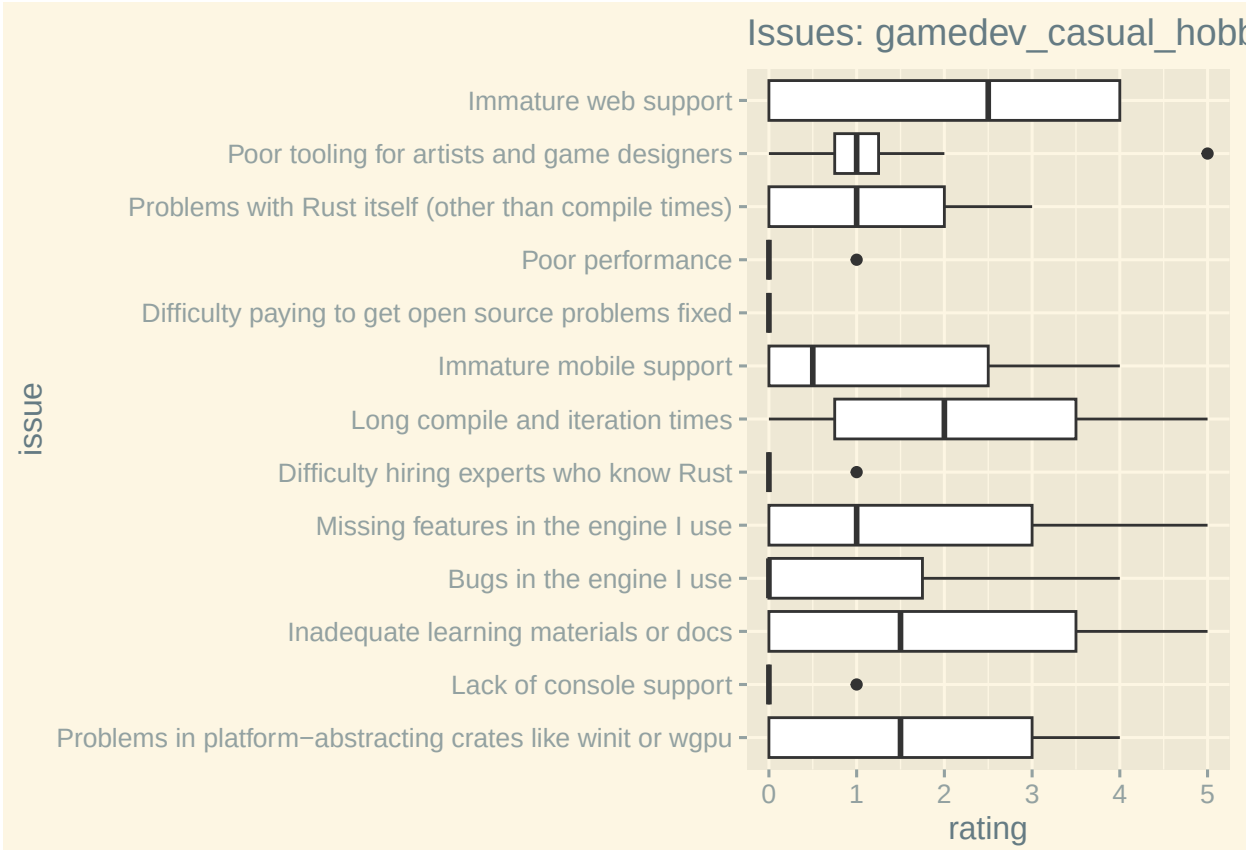

```
issue_labels <- c(
    "bad_iteration_time" = "Long compile and iteration times",
    "bad_rust" = "Problems with Rust itself (other than compile times)",
    "bad_abstraction" = "Problems in platform-abstracting crates like winit or wgpu",
    "bad_docs" = "Inadequate learning materials or docs",
    "bad_tooling" = "Poor tooling for artists and game designers",
    "bad_paying_for_bugs" = "Difficulty paying to get open source problems fixed",
    "bad_console" = "Lack of console support",
    "bad_mobile" = "Immature mobile support",
    "bad_web" = "Immature web support",
    "bad_engine_bugs" = "Bugs in the engine I use",
    "bad_engine_features" = "Missing features in the engine I use",
    "bad_hiring" = "Difficulty hiring experts who know Rust",
    "bad_performance" = "Poor performance"
  )

dat |>
  pivot_longer(cols = 4:16, names_to = "issue", values_to = "rating") |>
  ggplot(aes(x = issue, y = rating)) +
  geom_boxplot() +
  ggtitle("Issues in the ecosystem") +
  scale_x_discrete(labels = issue_labels) +
  coord_flip()
```

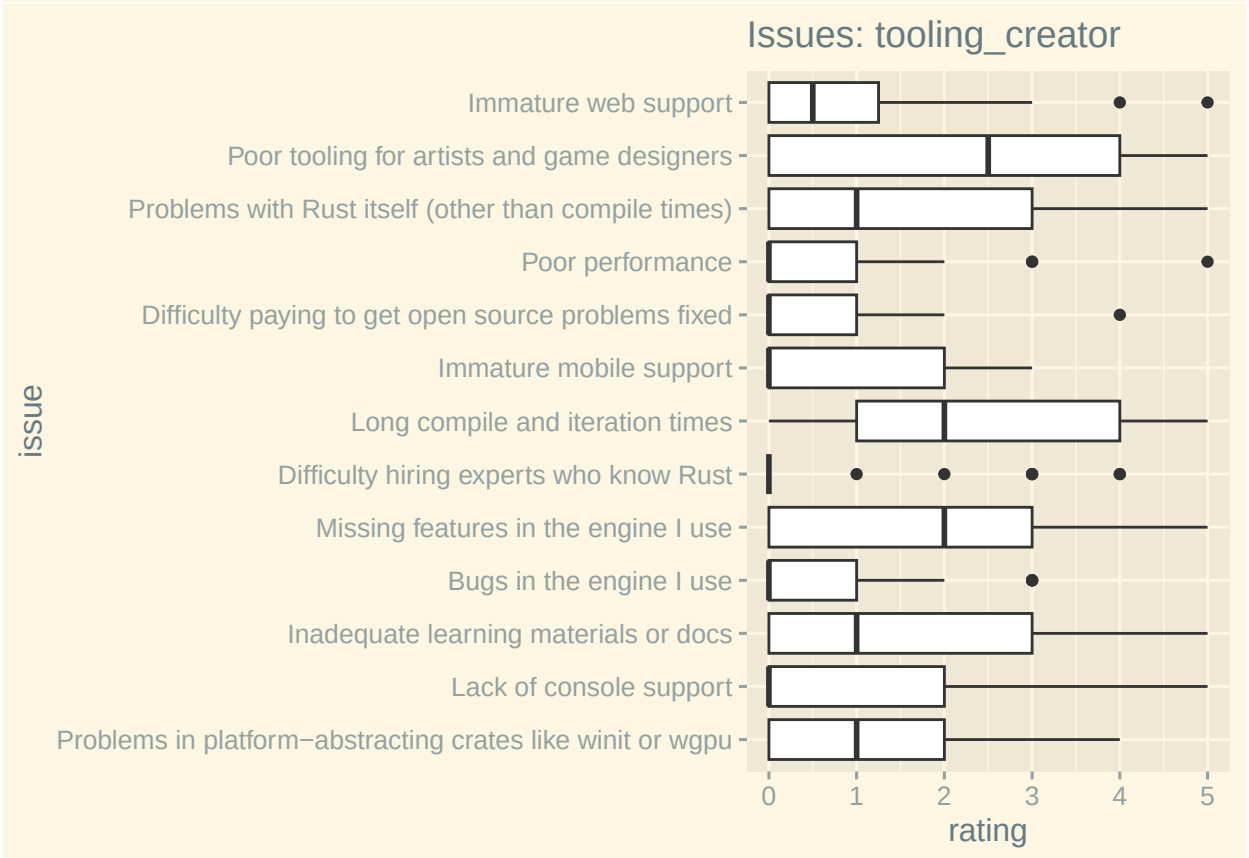


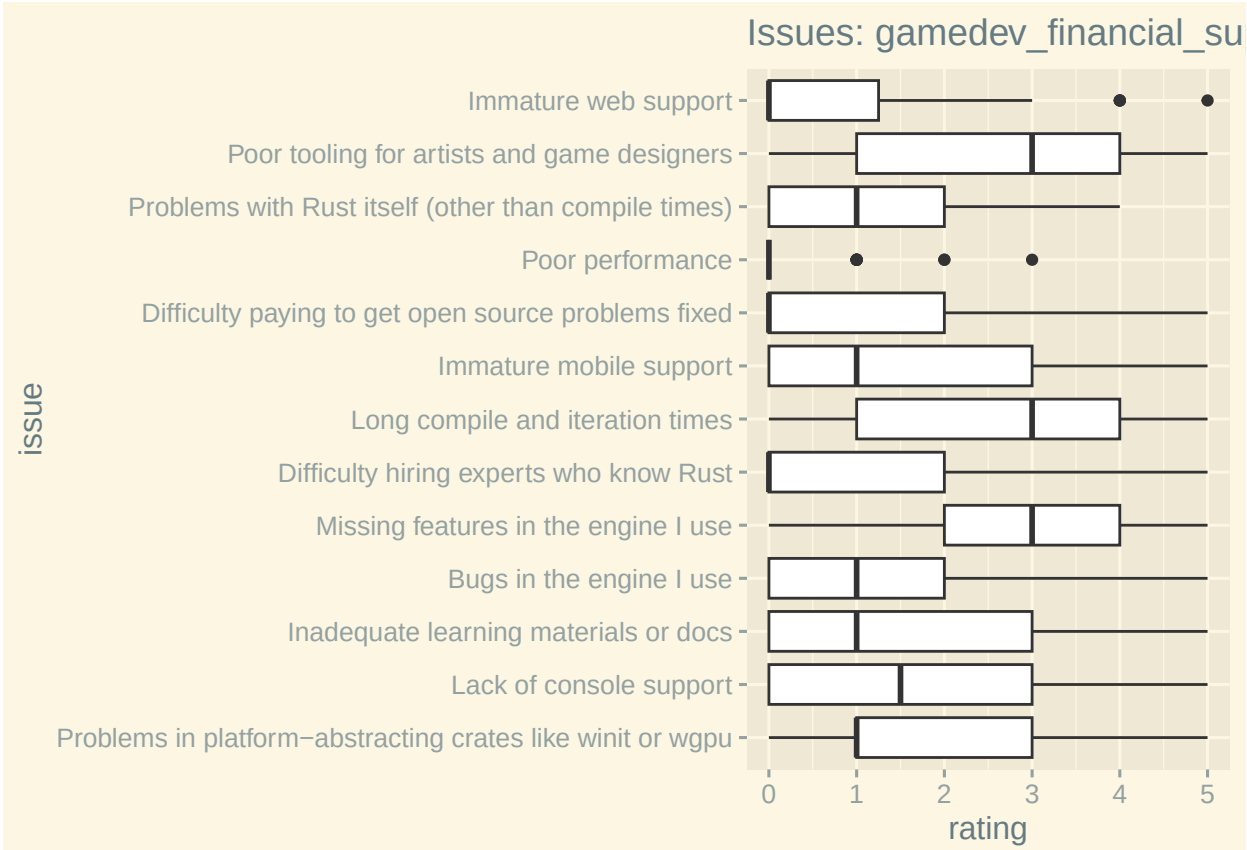
Issues by Use-Case

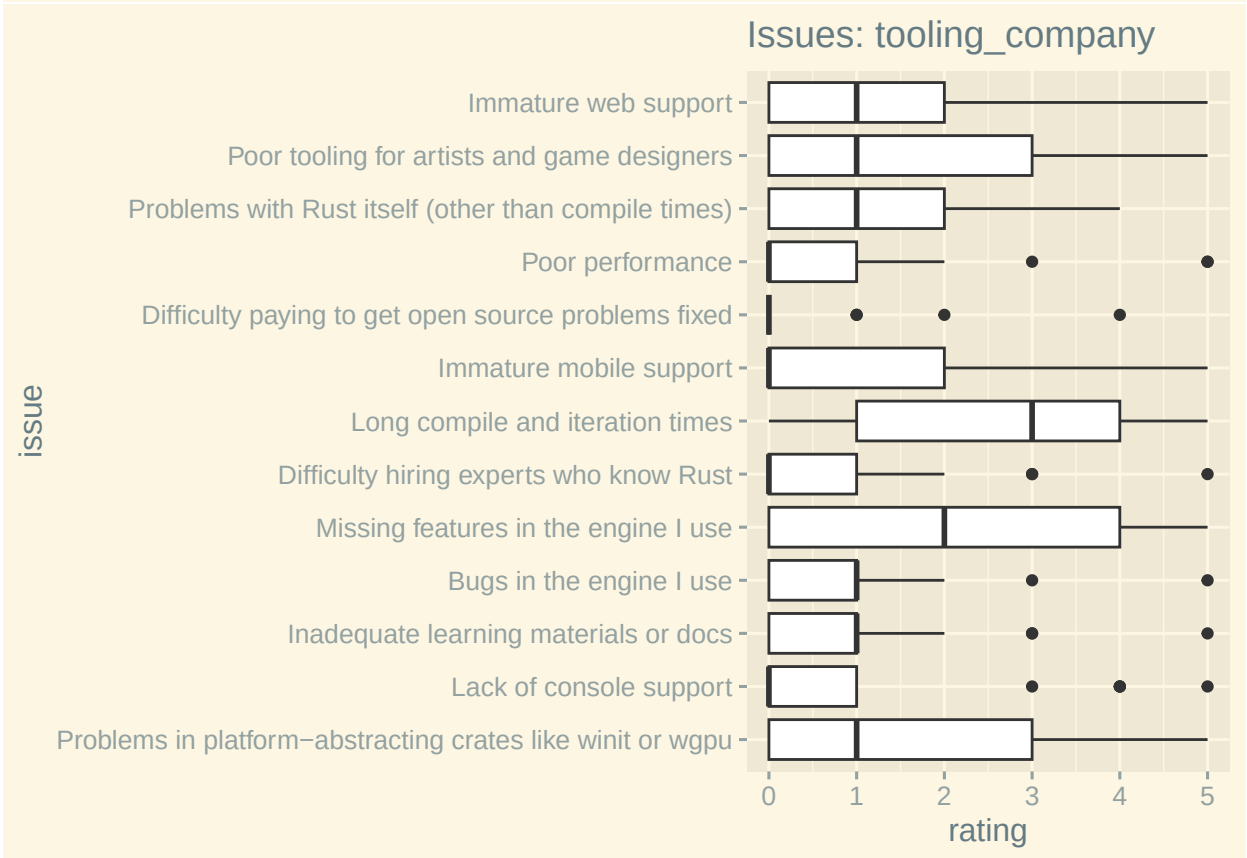
Plot all issues by use case

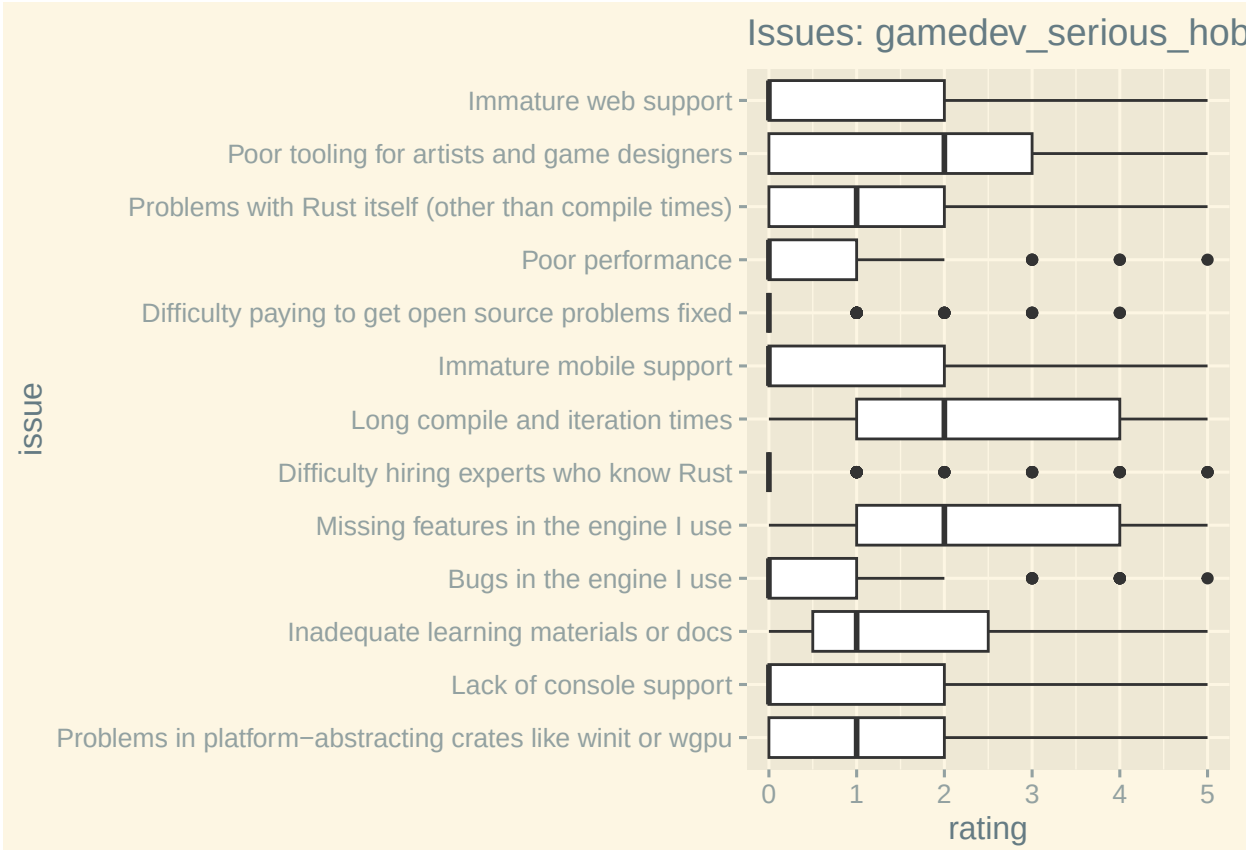
```
for (use_case in levels(dat$usage)) {
  print(dat |>
  filter(usage == use_case) |>
  pivot_longer(cols = 4:16, names_to = "issue", values_to = "rating") |>
  ggplot(aes(x = issue, y = rating)) +
  geom_boxplot() +
  ggtitle(glue("Issues: {use_case}")) +
  scale_x_discrete(labels = issue_labels) +
  coord_flip())
}
```

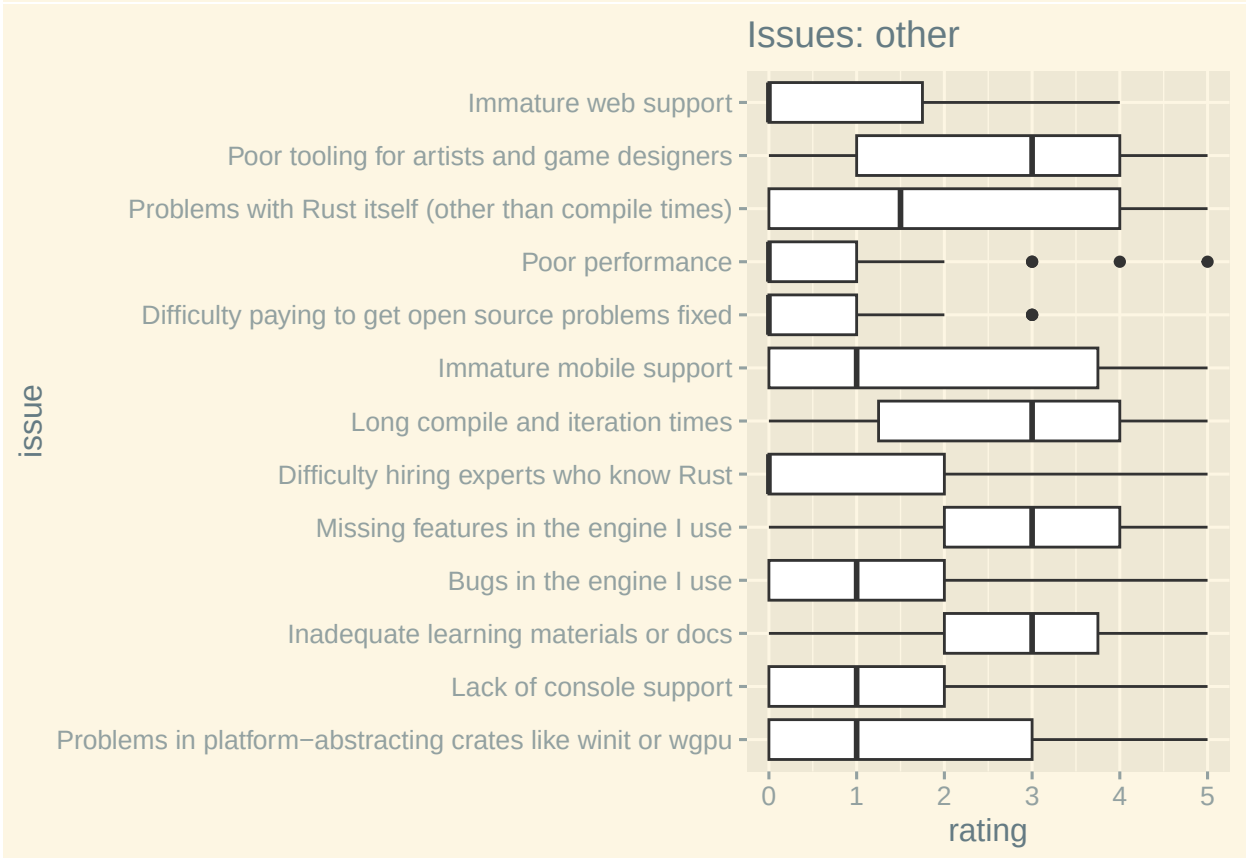

Issues: gamedev_casual_hobb
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
5
rating

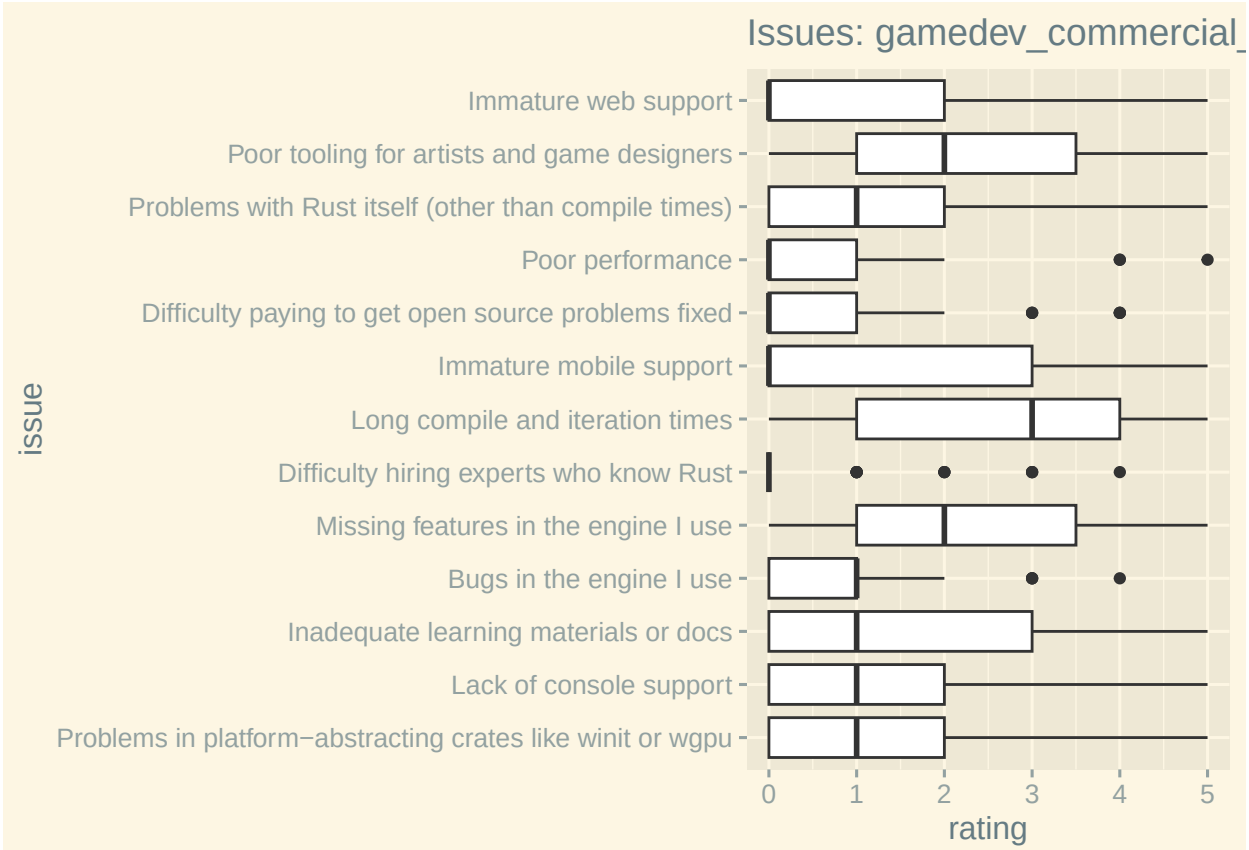

Issues: tooling_creator
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
5
rating

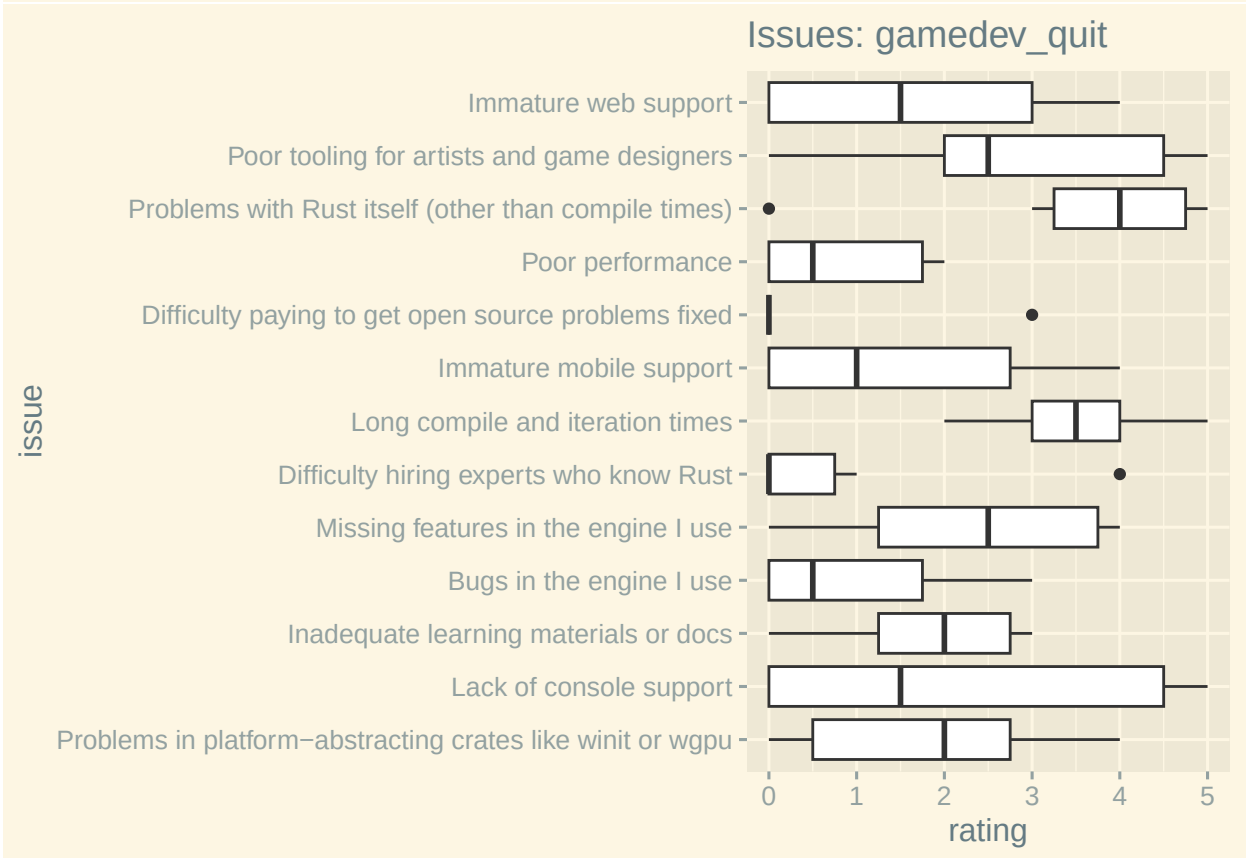

Issues: gamedev_financial_su
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
5
rating

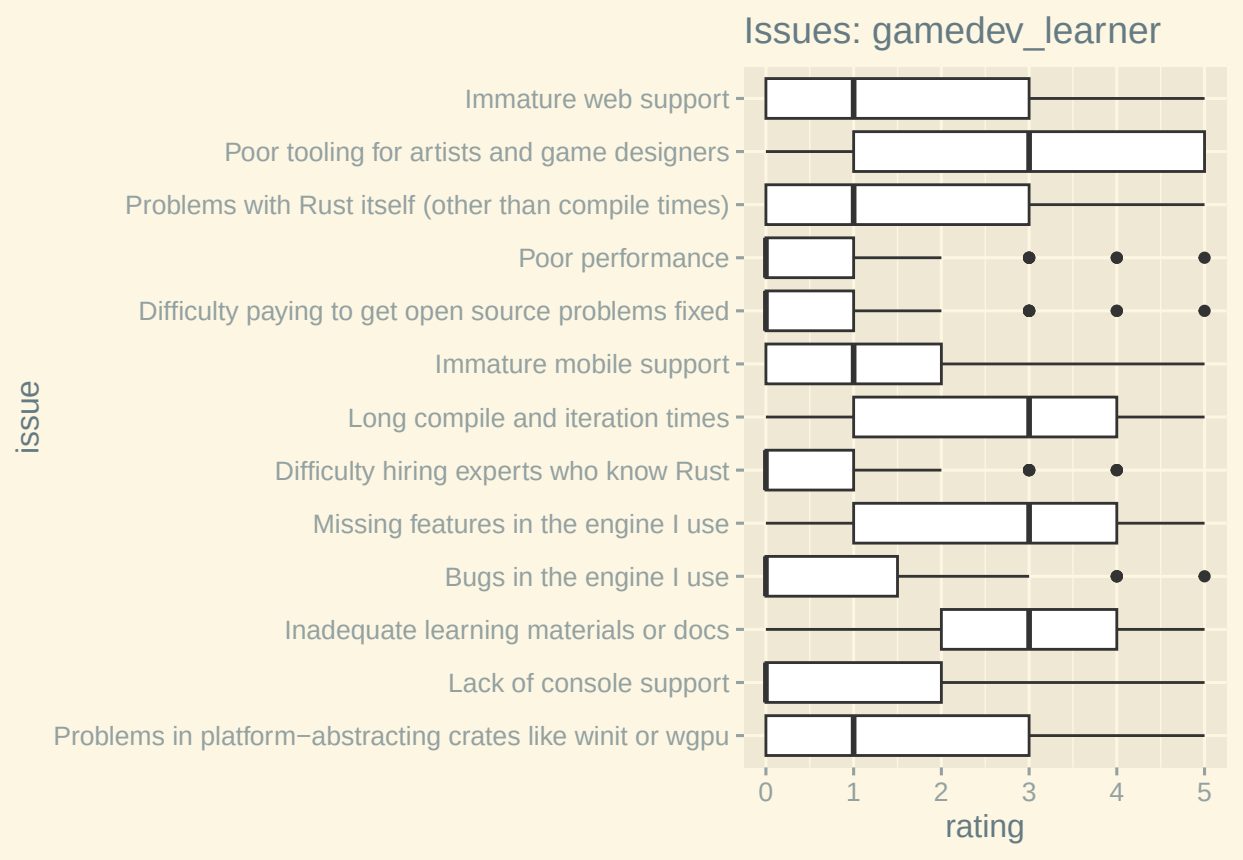

Issues: tooling_company
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
5
rating


Issues: gamedev_serious_hob
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
5
rating


Issues: other
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
5
rating


Issues: gamedev_commercial_
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
5
rating


Issues: gamedev_quit
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
5
rating



ANODE (Analysis of deviance)

```
issues <- dat[,4:16] |>
  as.matrix()

for (issue in colnames(issues)) {
  olm <- clm(as.factor(issues[,issue]) ~ usage, data = dat)
  coefficients <- summary(olm)$coefficients
  olm_row_names <- rownames(coefficients)

  "-----" |> glue() |> print()
  "Issue: {issue}" |> glue() |> print()

  if (any(is.na(coefficients[,4]))) {
    "Skipping because of NA" |> glue() |> print()
    next
  }
  olm.anova <- anova(olm)
  p_value <- olm.anova$`Pr(>Chisq)`
  bonferroni <- 0.05 / ncol(issues)

  "p_value: {p_value}" |> glue() |> print()

  if (p_value > bonferroni) {
    next
```

```
  }
  "Significant!" |> glue() |> print()

}
```

```
## -----
## Issue: bad_iteration_time
## p_value: 0.683646252627446
## -----
## Issue: bad_rust
## p_value: 0.00533530901276412
## -----
## Issue: bad_abstraction
## p_value: 0.215669311391801
## -----
## Issue: bad_docs
## p_value: 9.61322969216496e-11
## Significant!
## -----
## Issue: bad_tooling
## p_value: 0.0279869852485465
```

```
## Warning: (1) Hessian is numerically singular: parameters are not uniquely determined
## In addition: Absolute convergence criterion was met, but relative criterion was not met
```

```
## -----
## Issue: bad_paying_for_bugs
## Skipping because of NA
## -----
## Issue: bad_console
## p_value: 0.00959732945850934
## -----
## Issue: bad_mobile
## p_value: 0.288062177879454
## -----
## Issue: bad_web
## p_value: 0.16469495232012
## -----
## Issue: bad_engine_bugs
## p_value: 0.477727743175982
## -----
## Issue: bad_engine_features
## p_value: 0.155646288525414
## -----
## Issue: bad_hiring
## p_value: 0.170740384641027
## -----
## Issue: bad_performance
## p_value: 0.253239698737779
```

Using a Bonferroni correction, the only issue that is significantly correlated with the use-case is `bad docs`: "Inadequate learning materials or docs".

Post-Hoc Analysis

```
olm <- clm(as.factor(bad_docs) ~ usage, data = dat)
summary(olm) |> print()
```

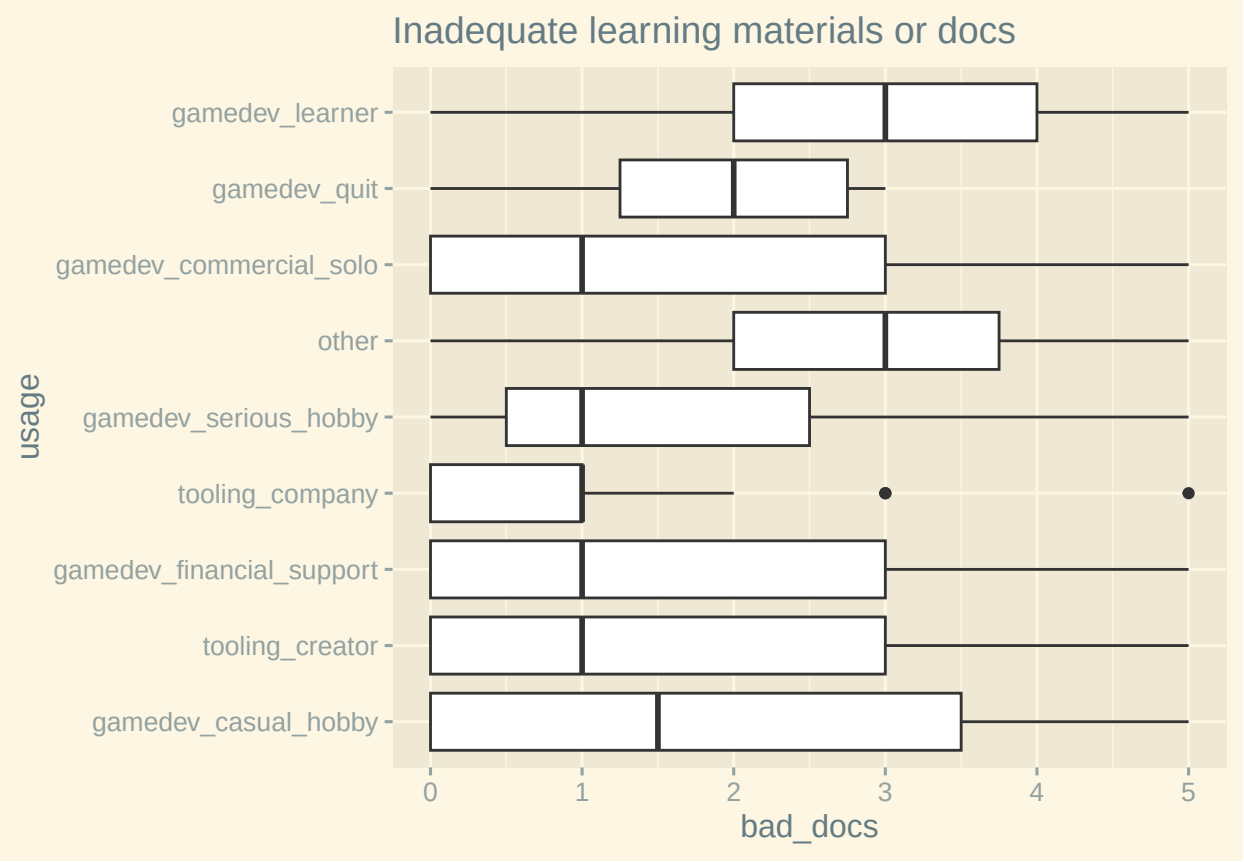
```
## formula: as.factor(bad_docs) ~ usage
## data:    dat
##
##  link  threshold nobs logLik  AIC     niter max.grad cond.H
##  logit flexible  410  -683.39 1392.79 4(0)  3.30e-08 2.2e+03
##
## Coefficients:
##                                Estimate Std. Error z value Pr(>|z|)
## usagetooling_creator            -0.3570     0.7862  -0.454   0.6498
## usagegamedev_financial_support  -0.1317     0.7768  -0.170   0.8653
## usagetooling_company            -0.9037     0.8018  -1.127   0.2597
## usagegamedev_serious_hobby      -0.1394     0.7387  -0.189   0.8503
## usageother                       1.0884     0.7844   1.388   0.1653
## usagegamedev_commercial_solo    -0.3184     0.7657  -0.416   0.6776
## usagegamedev_quit                0.1695     0.9809   0.173   0.8628
## usagegamedev_learner             1.4147     0.7501   1.886   0.0593 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Threshold coefficients:
##     Estimate Std. Error z value
## 0|1 -1.17012    0.72584  -1.612
## 1|2 -0.05315    0.72442  -0.073
## 2|3  0.80794    0.72574   1.113
## 3|4  1.66435    0.73006   2.280
## 4|5  2.54944    0.73823   3.453
```

```
confint(olm) |> print()
```

```
##                                      2.5 %    97.5 %
## usagetooling_creator           -1.91741045 1.1982617
## usagegamedev_financial_support -1.67344003 1.4063924
## usagetooling_company           -2.49624471 0.6794119
## usagegamedev_serious_hobby     -1.60887810 1.3271746
## usageother                     -0.46467027 2.6438360
## usagegamedev_commercial_solo   -1.83959789 1.1983775
## usagegamedev_quit              -1.78189940 2.0977977
## usagegamedev_learner           -0.07240164 2.9053679
```

Although it seems like the documentation issue correlates with the use-case, the concrete coefficients are not significant. Bummer. Can I offer you a plot instead?

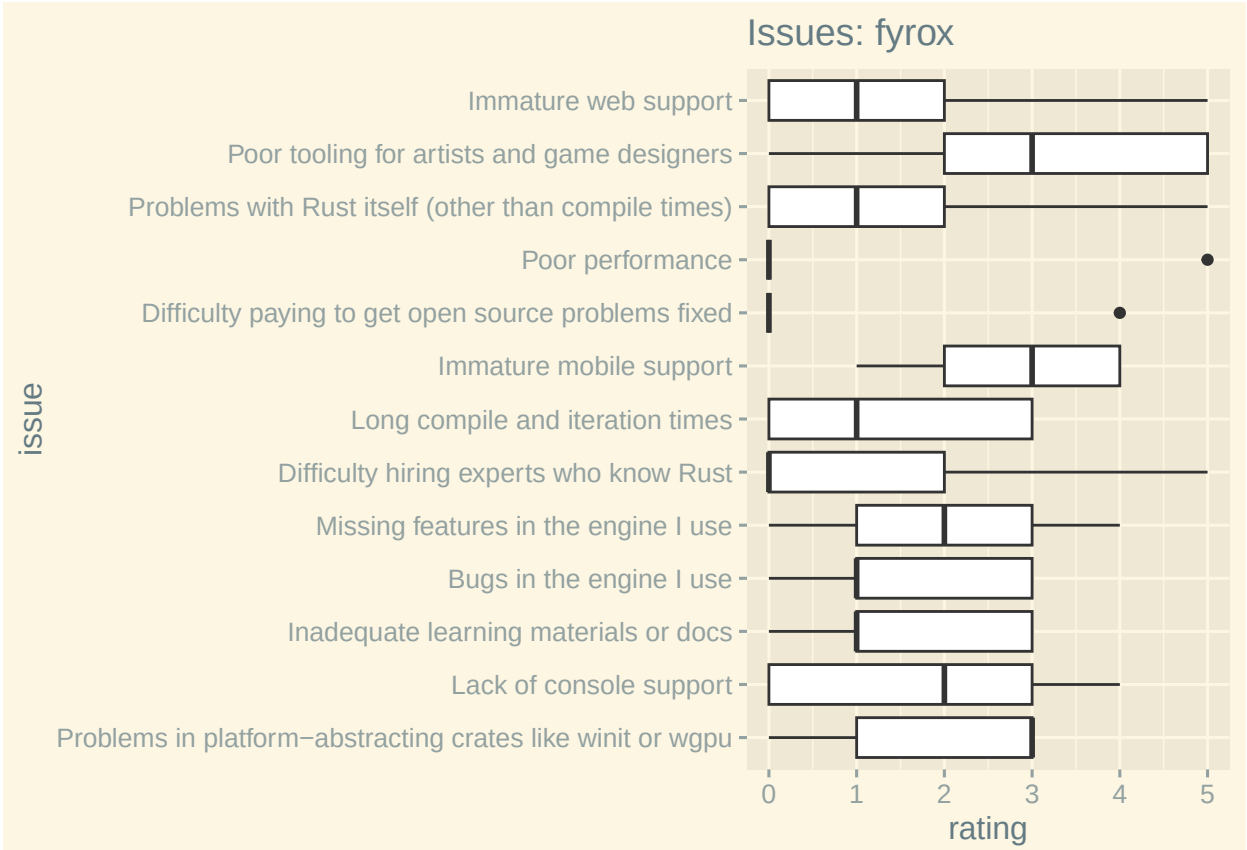
```
dat |>
  ggplot(aes(x = bad_docs, y = usage)) +
  geom_boxplot() +
  ggtitle("Inadequate learning materials or docs")
```

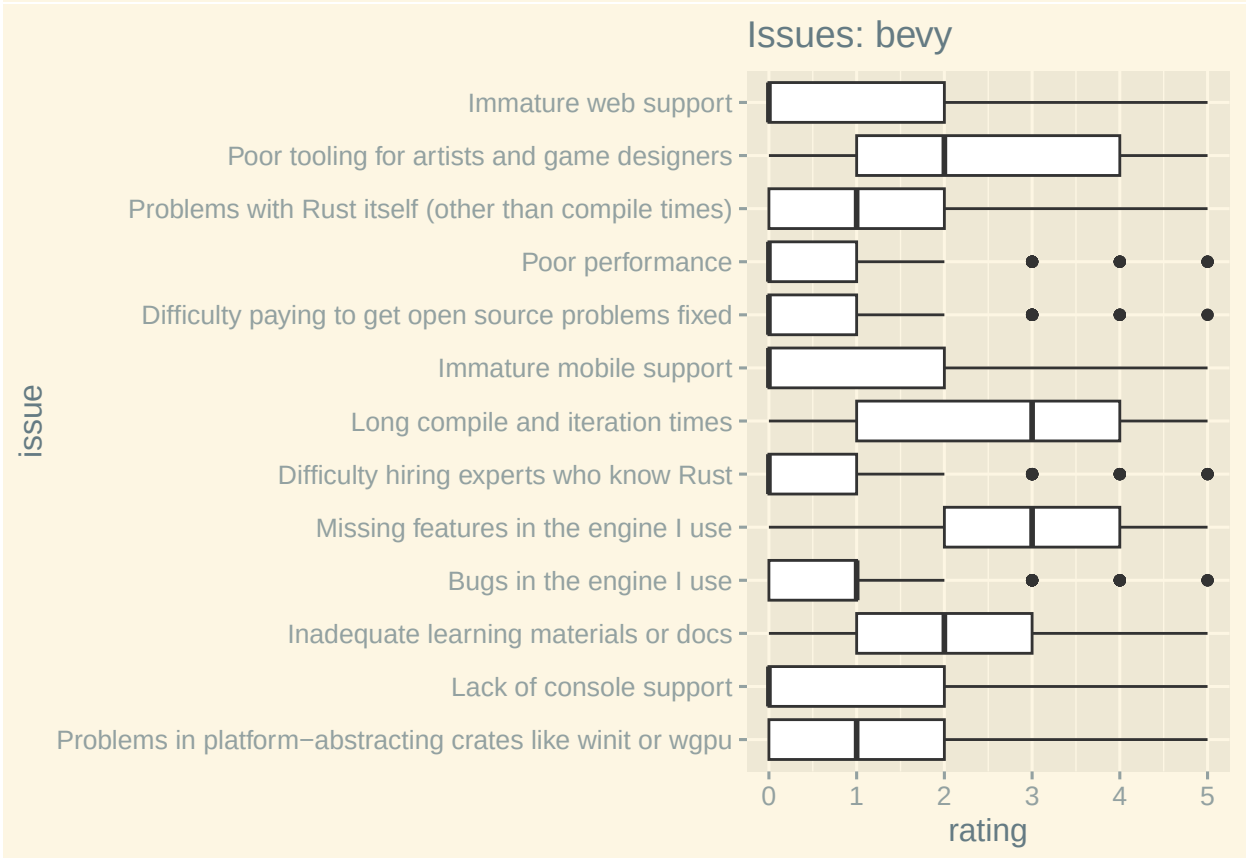


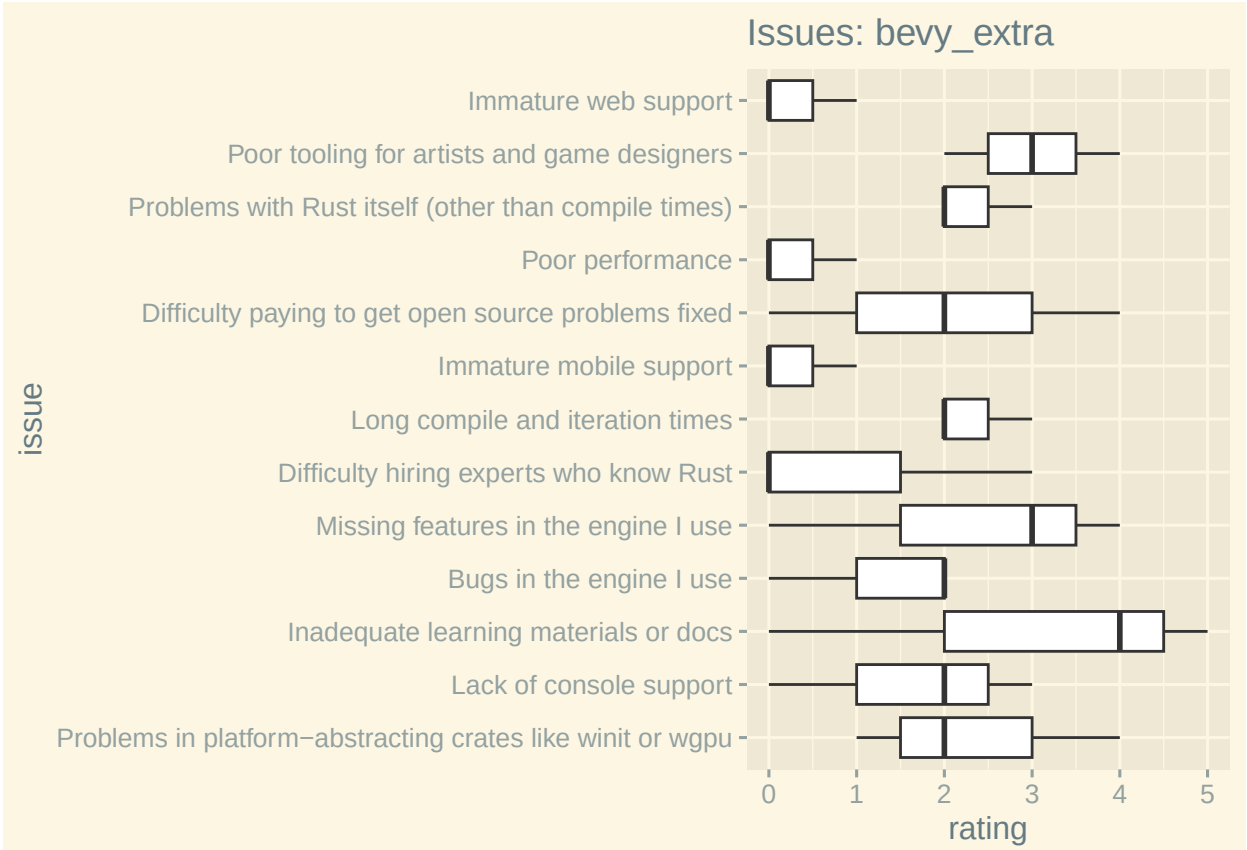
Issues by Engine

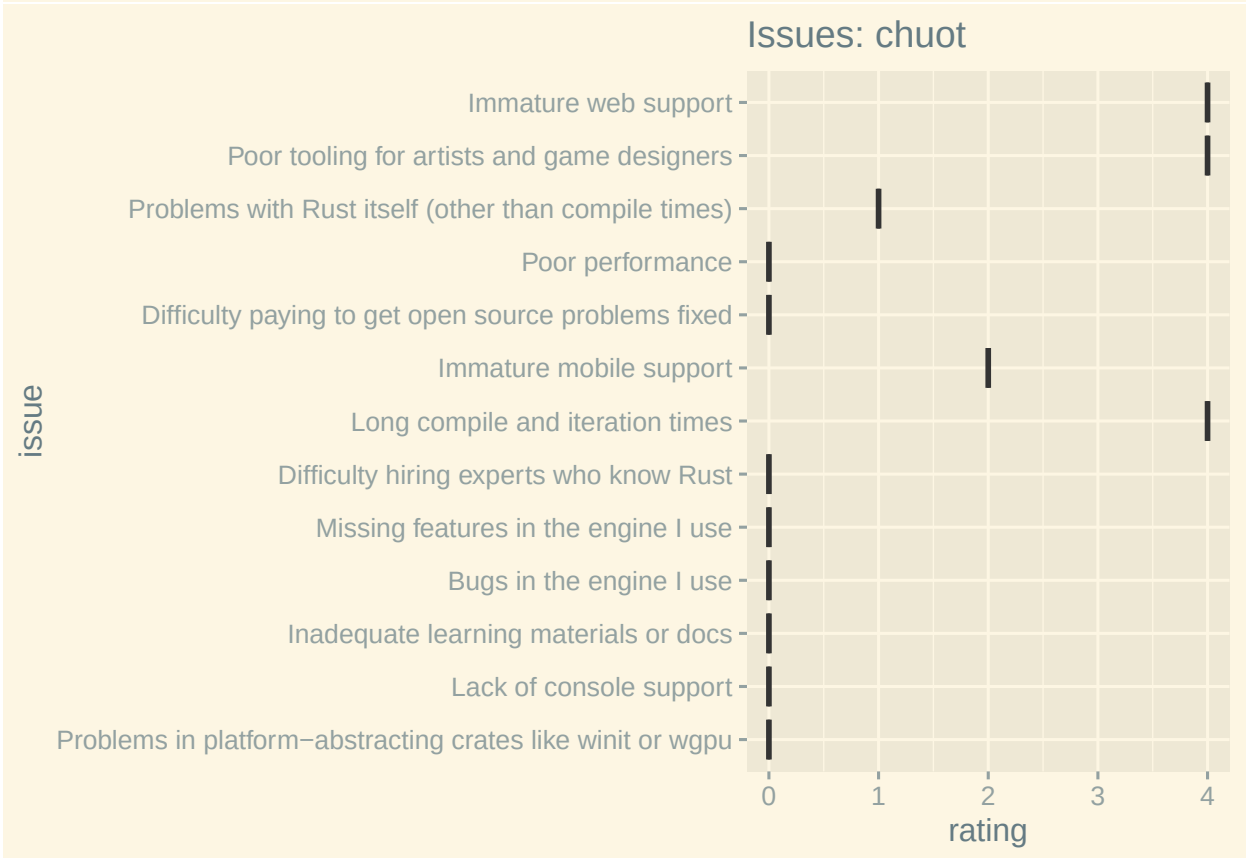
Plot all issues by engine

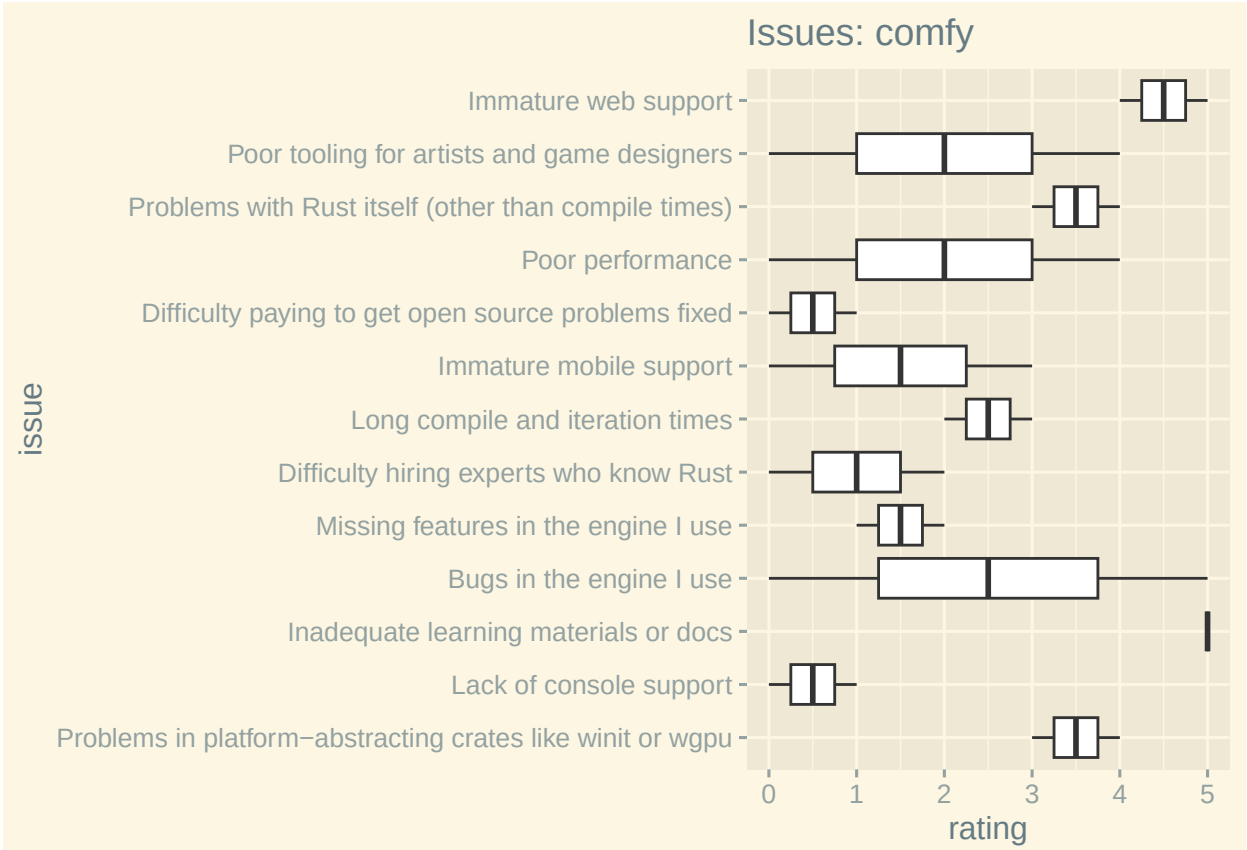
```
for (engine_level in levels(dat$engine)) {
  print(dat |>
  filter(engine == engine_level) |>
  pivot_longer(cols = 4:16, names_to = "issue", values_to = "rating") |>
  ggplot(aes(x = issue, y = rating)) +
  geom_boxplot() +
  ggtitle(glue("Issues: {engine_level}")) +
  scale_x_discrete(labels = issue_labels) +
  coord_flip())
}
```

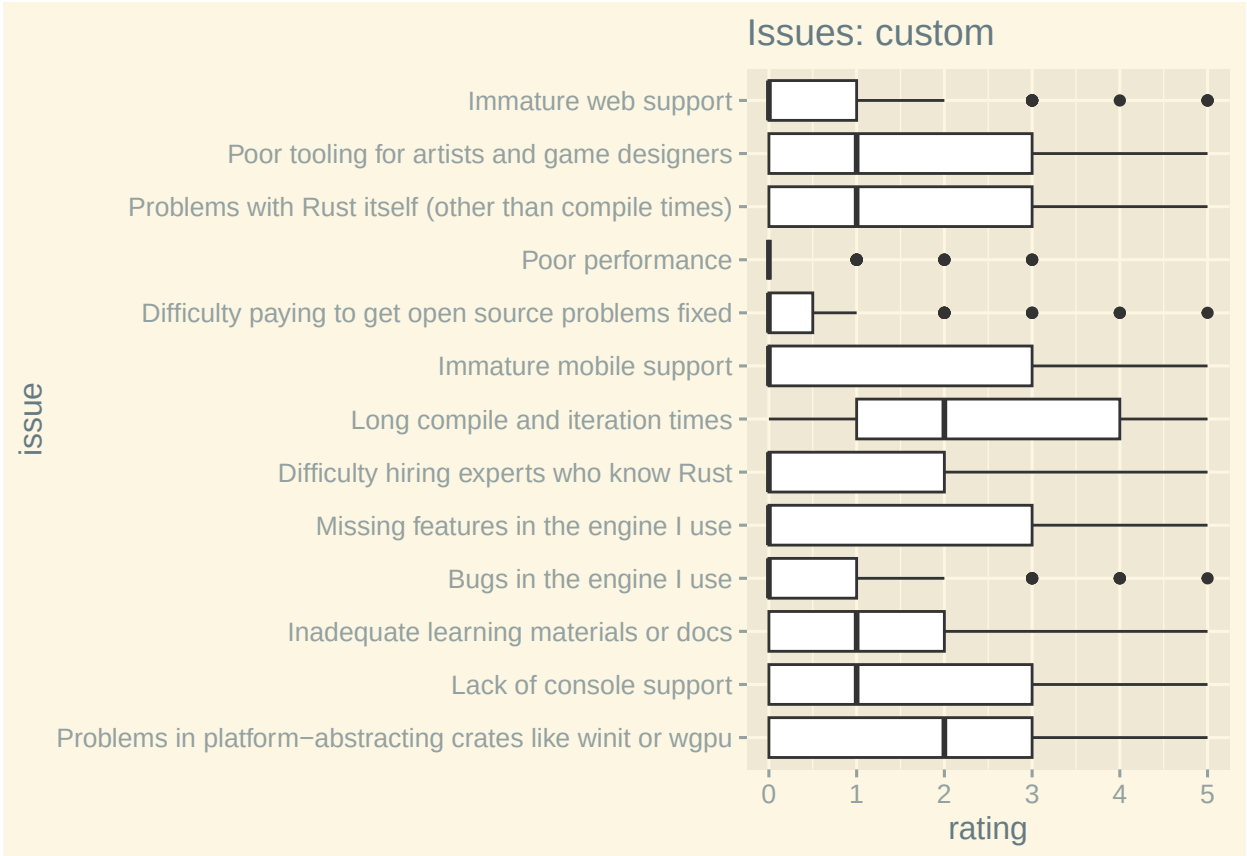

Issues: fyrox
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
5
rating

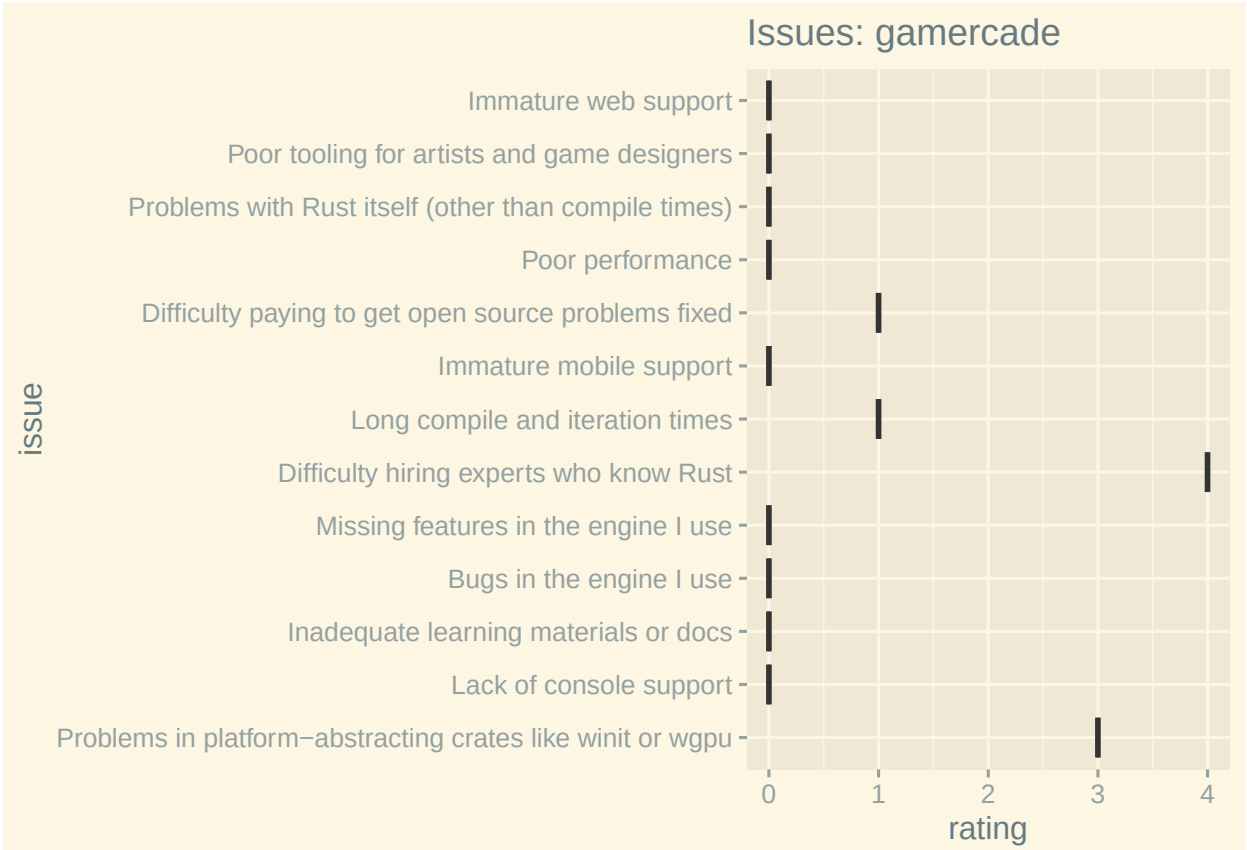

Issues: bevy
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
5
rating

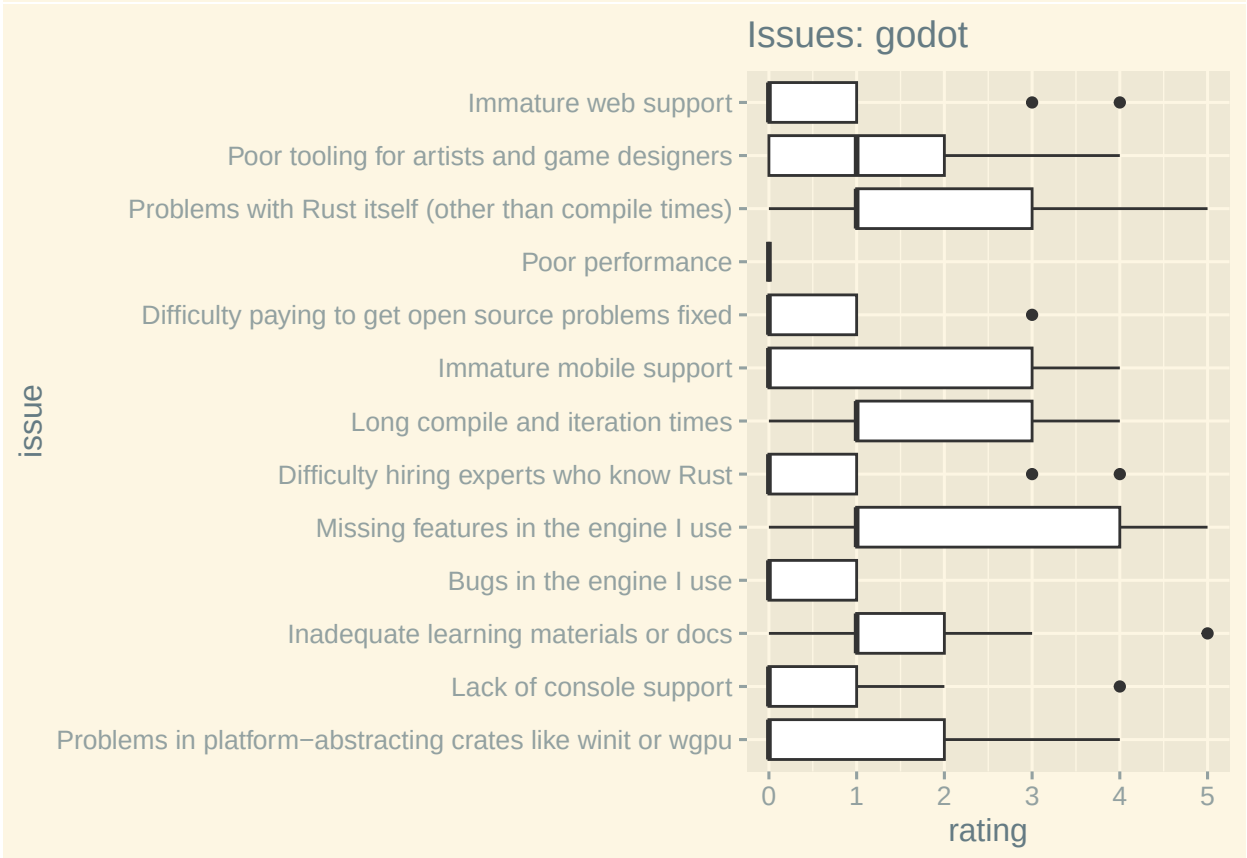

Issues: bevy_extra
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
5
rating

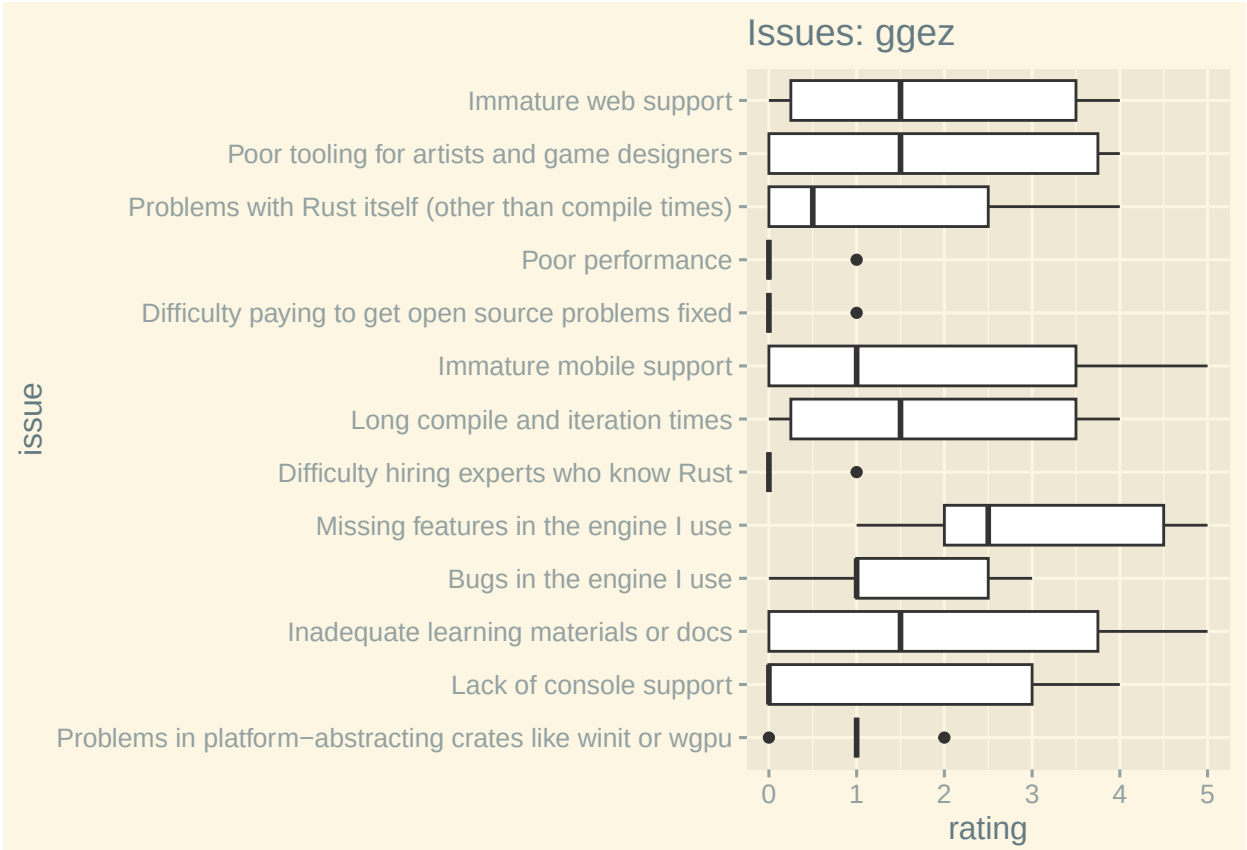

Issues: chuot
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
rating

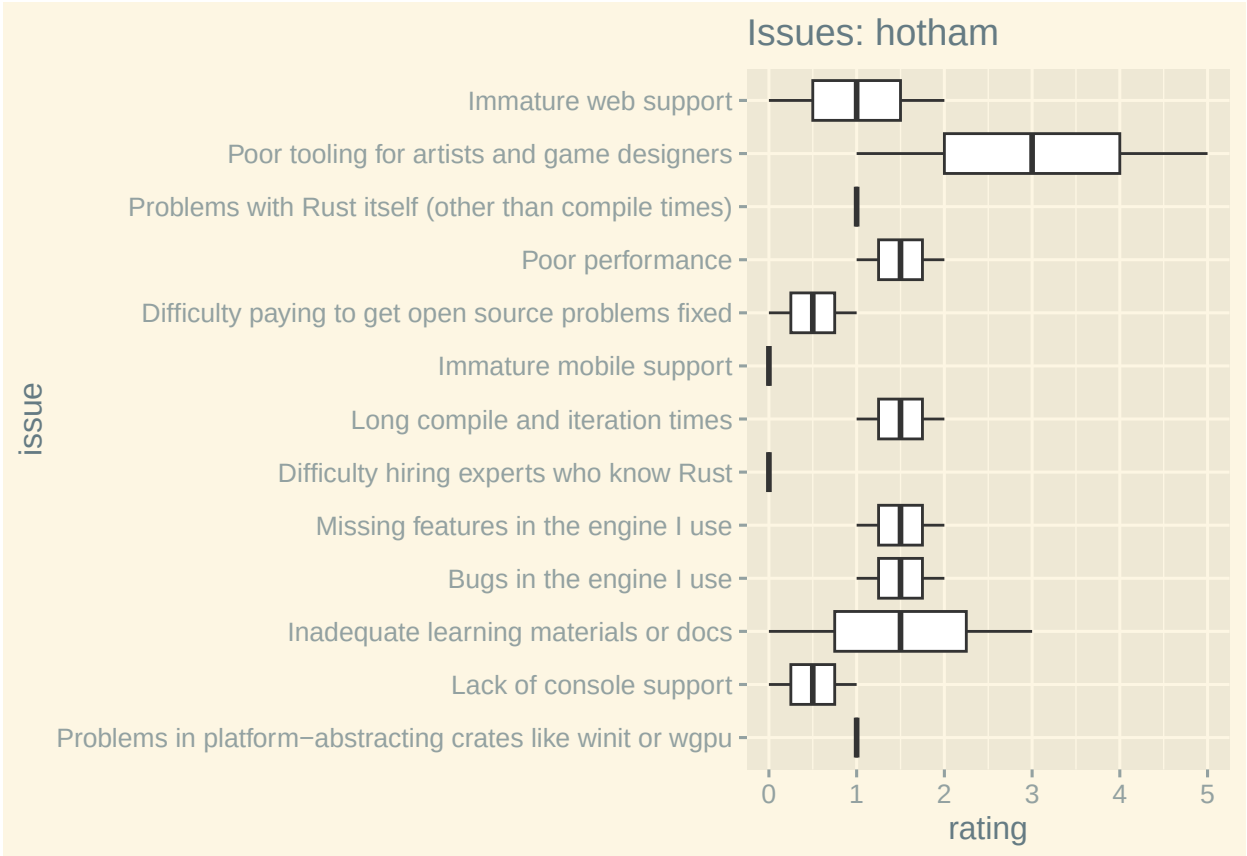

Issues: comfy
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
5
rating

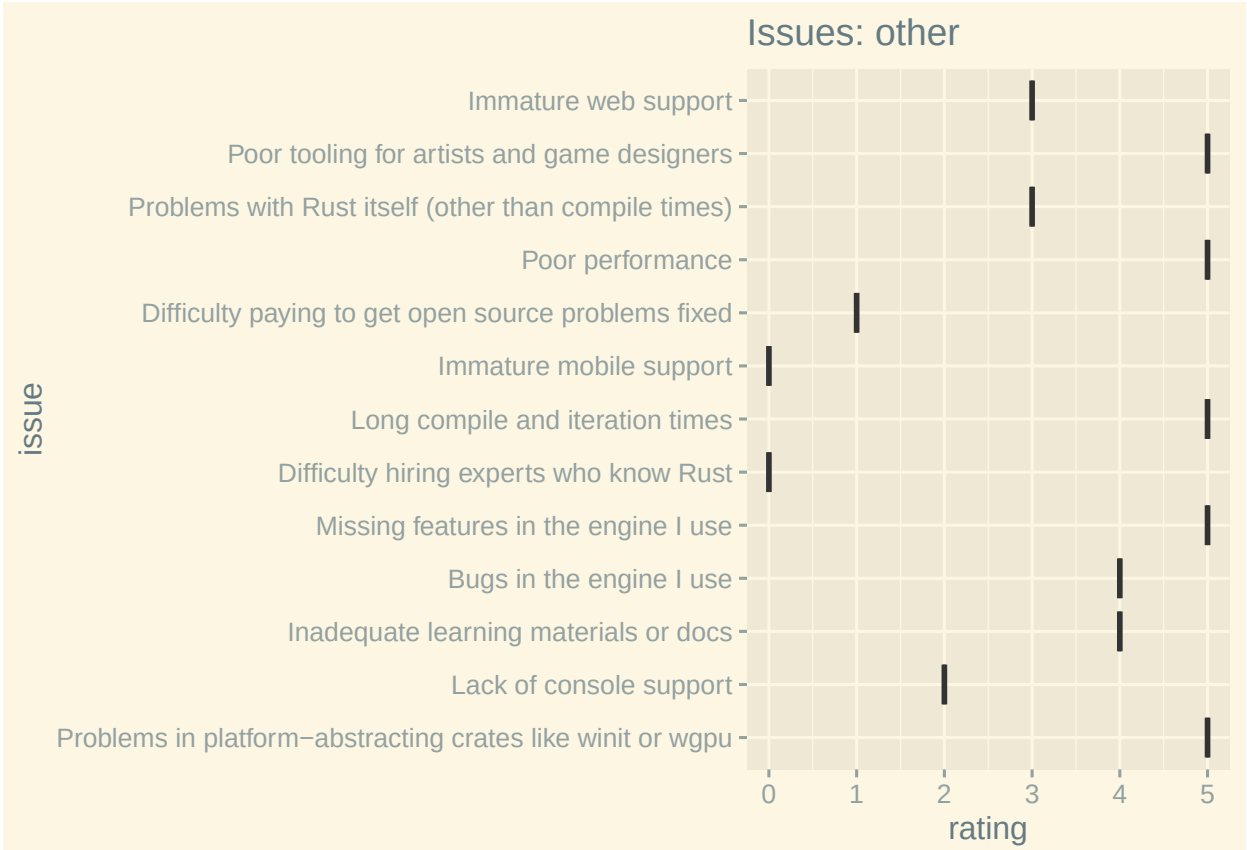

Issues: custom
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
5
rating

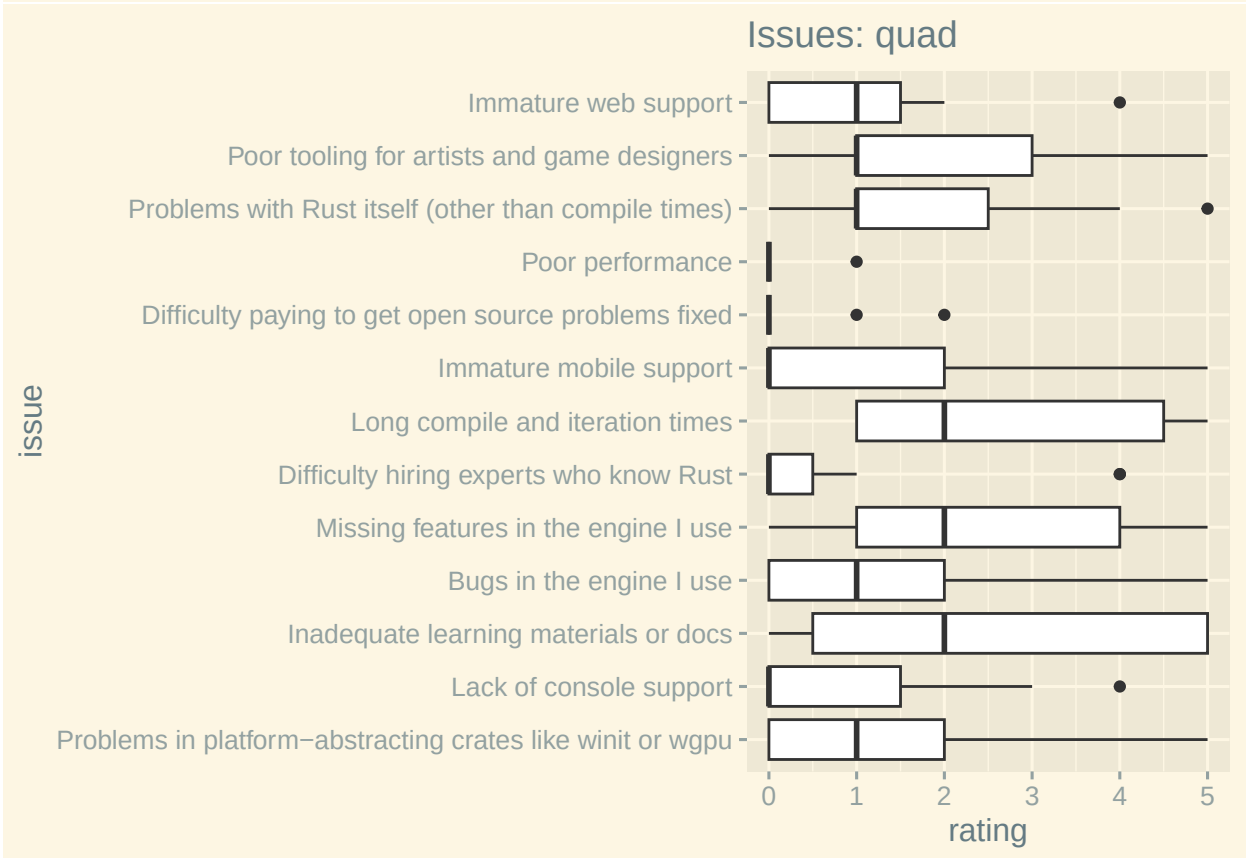

Issues: gamercade
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
rating

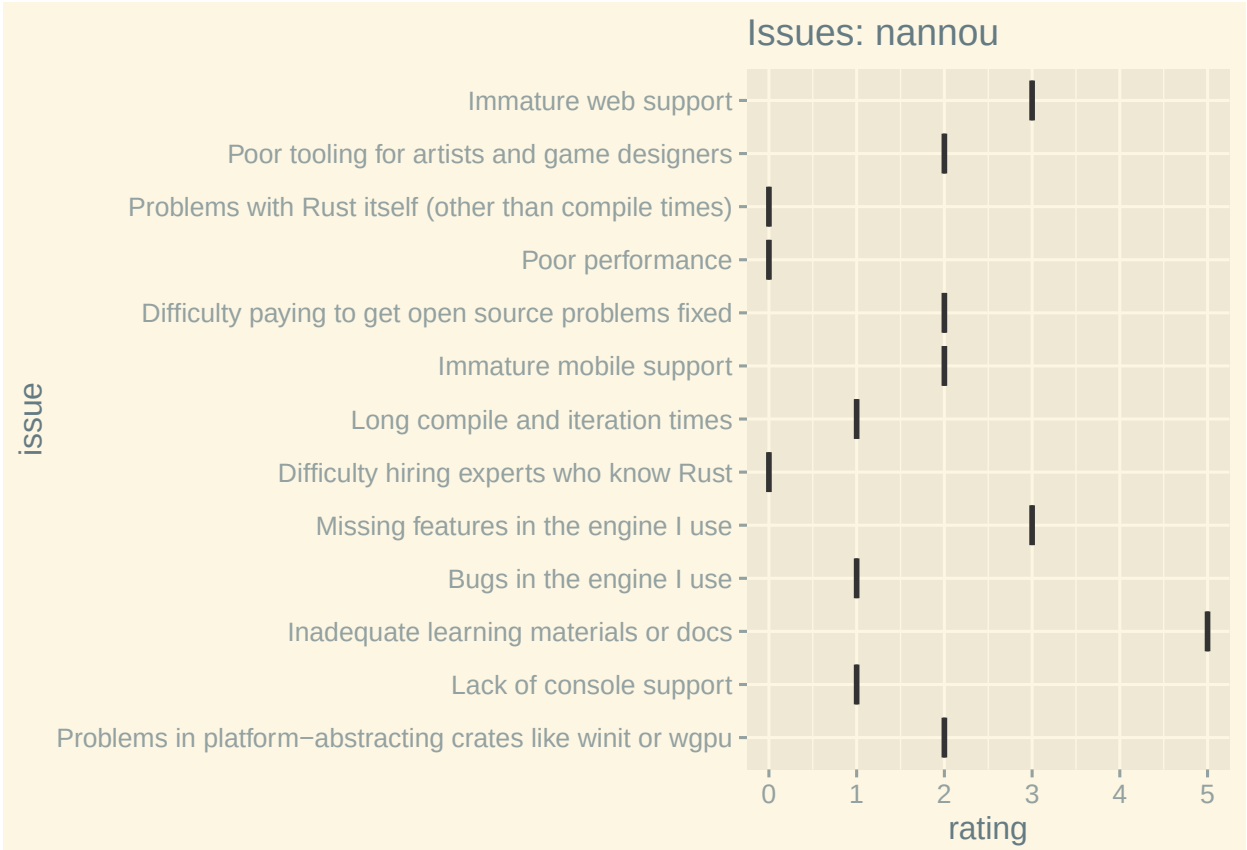

Issues: godot
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
5
rating

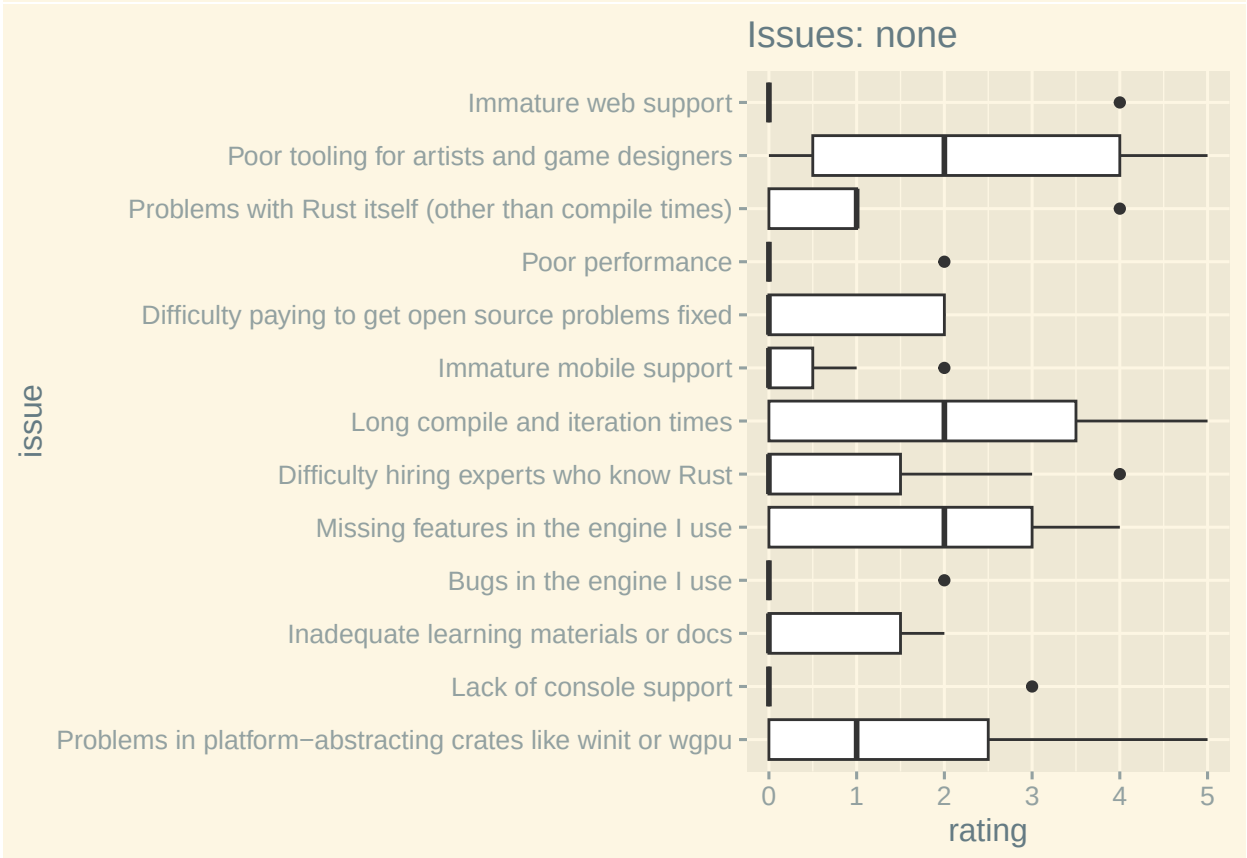

Issues: ggez
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
5
rating

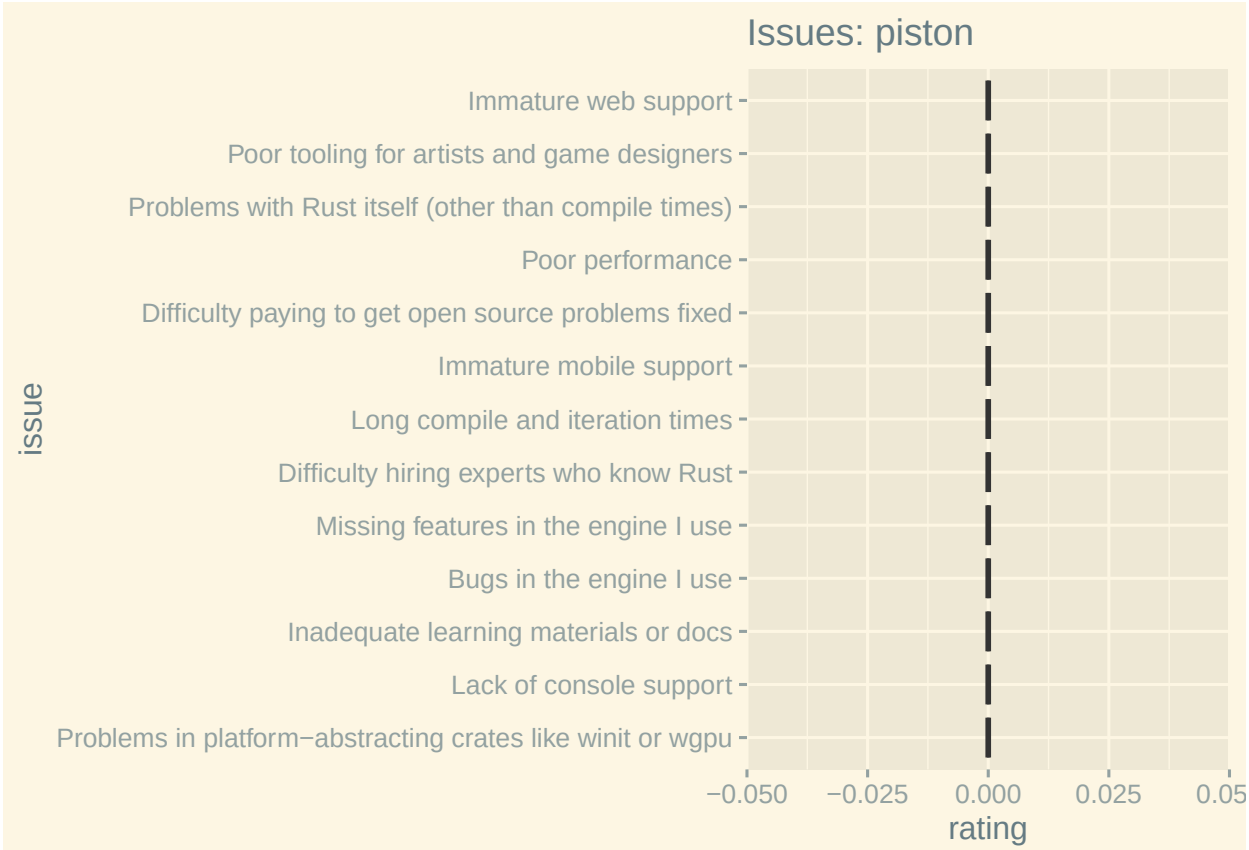

Issues: hotham
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
5
rating

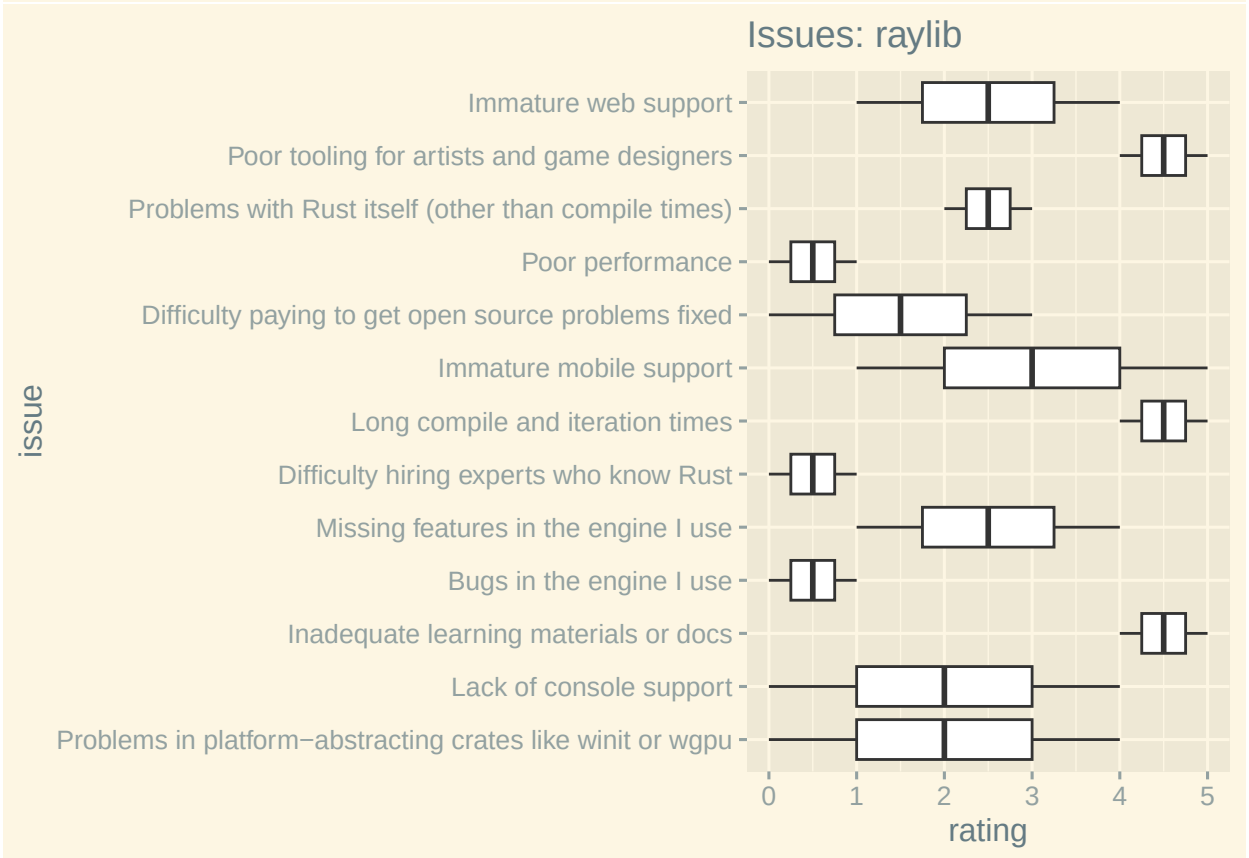

Issues: other
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
5
rating

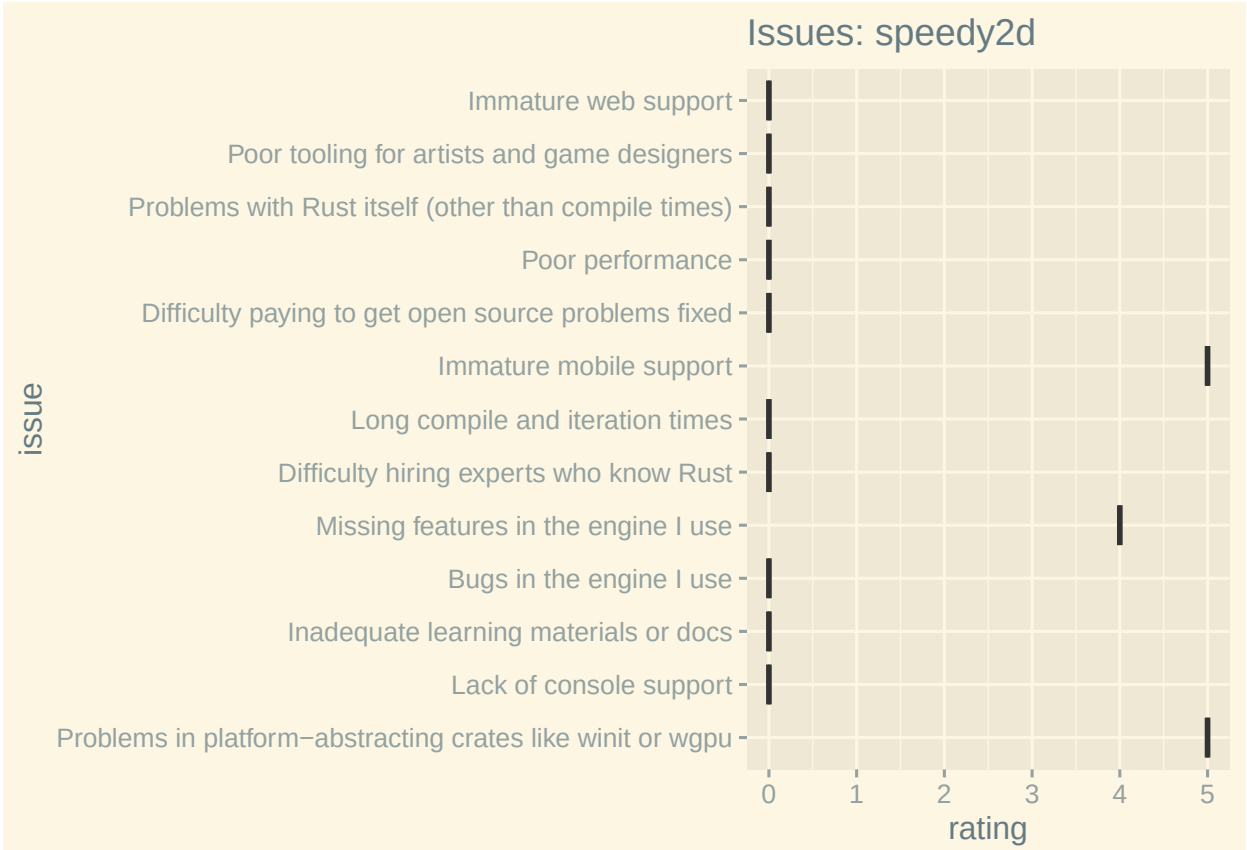

Issues: quad
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
5
rating

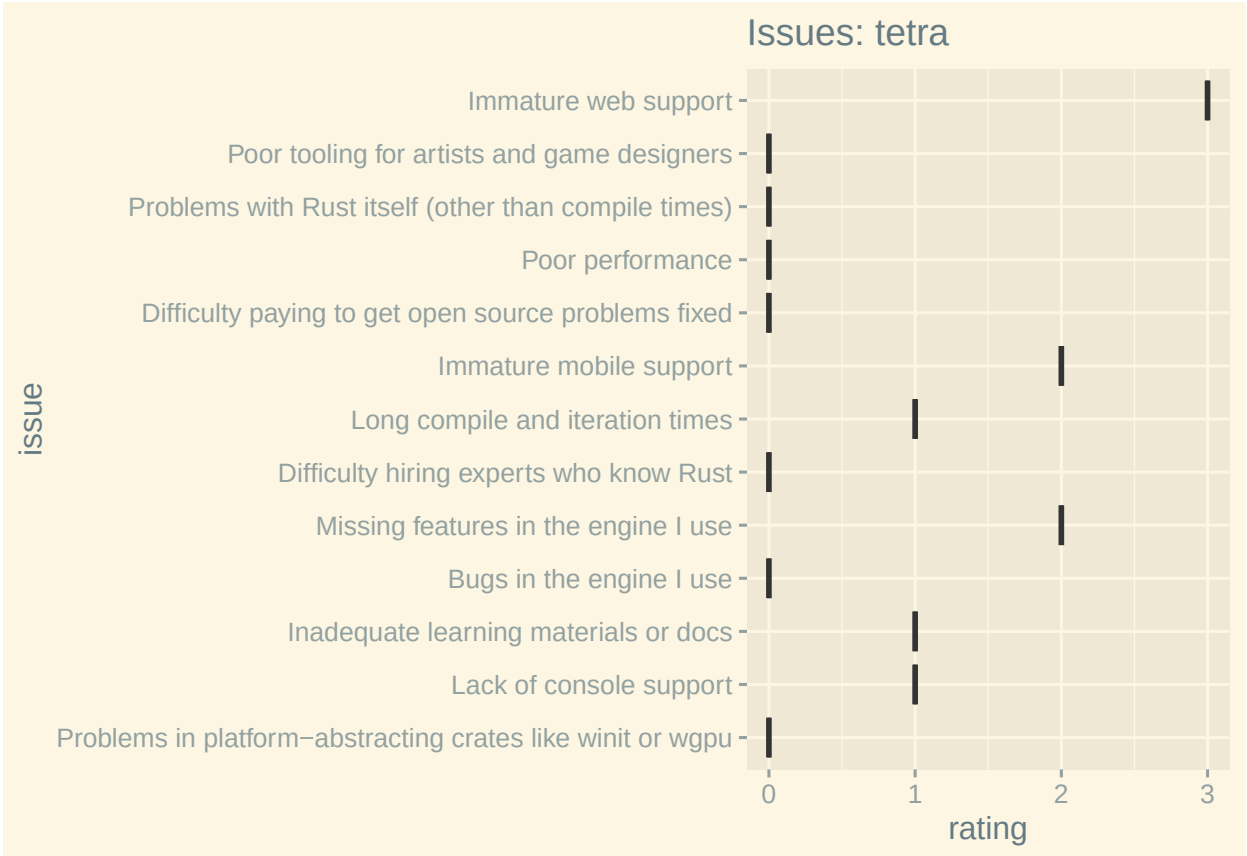

Issues: nannou
issue
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
0
1
2
3
4
5
rating


Issues: none
issue
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
0
1
2
3
4
5
rating


Issues: piston
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
−0.050
−0.025
0.000
0.025
0.05
rating


Issues: raylib
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
5
rating


Issues: speedy2d
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
4
5
rating


Issues: tetra
Immature web support
Poor tooling for artists and game designers
Problems with Rust itself (other than compile times)
Poor performance
Difficulty paying to get open source problems fixed
Immature mobile support
Long compile and iteration times
Difficulty hiring experts who know Rust
Missing features in the engine I use
Bugs in the engine I use
Inadequate learning materials or docs
Lack of console support
Problems in platform−abstracting crates like winit or wgpu
issue
0
1
2
3
rating

ANODE (Analysis of deviance)

```
# remove data with too few answers
dat.cleaned <- dat |>
  dplyr::group_by(engine) |>
  dplyr::filter(n() > 9) |>
  dplyr::ungroup()


issues <- dat.cleaned[,4:16] |>
  as.matrix()

for (issue in colnames(issues)) {
  olm <- clm(as.factor(issues[,issue]) ~ engine, data = dat.cleaned)

  coefficients <- summary(olm)$coefficients
  olm_row_names <- rownames(coefficients)

  "-----" |> glue() |> print()
  "Issue: {issue}" |> glue() |> print()

  if (any(is.na(coefficients[,4]))) {
    "Skipping because of NA" |> glue() |> print()
    next
  }
  olm.anova <- anova(olm)
  p_value <- olm.anova$`Pr(>Chisq)`
  bonferroni <- 0.05 / ncol(issues)

  "p_value: {p_value}" |> glue() |> print()

  if (p_value > bonferroni) {
    next
  }
  "Significant!" |> glue() |> print()
}
```

```
## -----
## Issue: bad_iteration_time
## p_value: 0.459302044596087
## -----
## Issue: bad_rust
## p_value: 0.0136107592878821
## -----
## Issue: bad_abstraction
## p_value: 0.000520490937872782
## Significant!
## -----
## Issue: bad_docs
## p_value: 0.000148578611972274
## Significant!
## -----
## Issue: bad_tooling
## p_value: 0.0449805895233615
```

```
## -----
## Issue: bad_paying_for_bugs
## p_value: 0.444014451632465
## -----
## Issue: bad_console
## p_value: 0.0269670312996515
## -----
## Issue: bad_mobile
## p_value: 0.908253835147806
## -----
## Issue: bad_web
## p_value: 0.61167863791296
## -----
## Issue: bad_engine_bugs
## p_value: 0.00465207528973174
## -----
## Issue: bad_engine_features
## p_value: 2.19796912873244e-08
## Significant!
## -----
## Issue: bad_hiring
## p_value: 0.295050068602354
## -----
## Issue: bad_performance
## p_value: 0.0131730938264978
```

The significant correlations happen in the following issues: - bad_abstraction: "Problems in platform-abstracting crates like winit or wgpu" - bad_docs: "Inadequate learning materials or docs" - bad_engine_features: "Missing features in the engine I use"

Post-Hoc Analysis

```
olm <- clm(as.factor(bad_abstraction) ~ engine, data = dat.cleaned)
summary(olm) |> print()
```

```
## formula: as.factor(bad_abstraction) ~ engine
## data:    dat.cleaned
##
##  link  threshold nobs logLik  AIC     niter max.grad cond.H
##  logit flexible  367  -558.38 1130.77 5(0)  1.34e-10 8.4e+01
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## enginecustom  0.97600    0.25147   3.881 0.000104 ***
## enginequad    0.05999    0.49744   0.121 0.904003
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Threshold coefficients:
##     Estimate Std. Error z value
## 0|1  -0.2251     0.1155  -1.950
## 1|2   0.6584     0.1206   5.460
## 2|3   1.3630     0.1388   9.822
## 3|4   2.3425     0.1842  12.716
```

```
## 4|5   3.7320     0.3169  11.776
```

```
confint(olm) |> print()
```

```
##                   2.5 %  97.5 %
## enginecustom  0.4831048 1.47047
## enginequad   -0.9478182 1.02616
```

Compared to Bevy, custom engines are much more likely to have problems with platform-abstracting crates like winit or wgpu.

```
olm <- clm(as.factor(bad_docs) ~ engine, data = dat.cleaned)
summary(olm) |> print()
```

```
## formula: as.factor(bad_docs) ~ engine
## data:    dat.cleaned
##
##  link  threshold nobs logLik  AIC     niter max.grad cond.H
##  logit flexible  367  -632.48 1278.95 4(0)  2.92e-10 1.0e+02
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## enginecustom  -1.0576     0.2566  -4.121 3.78e-05 ***
## enginequad     0.2656     0.5429   0.489    0.625
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Threshold coefficients:
##     Estimate Std. Error z value
## 0|1  -1.5234     0.1432 -10.640
## 1|2  -0.4266     0.1166  -3.659
## 2|3   0.3986     0.1161   3.433
## 3|4   1.2006     0.1336   8.986
## 4|5   2.0745     0.1775  11.690
```

```
confint(olm) |> print()
```

```
##                   2.5 %    97.5 %
## enginecustom -1.5661799 -0.558329
## enginequad   -0.7930081  1.347727
```

Compared to Bevy, custom engines are much less likely to have problems with inadequate learning materials or docs.

```
olm <- clm(as.factor(bad_engine_features) ~ engine, data = dat.cleaned)
summary(olm) |> print()
```

```
## formula: as.factor(bad_engine_features) ~ engine
## data:    dat.cleaned
##
##  link  threshold nobs logLik  AIC     niter max.grad cond.H
##  logit flexible  367  -636.42 1286.84 4(0)  1.33e-09 8.7e+01
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## enginecustom  -1.6661     0.2812  -5.924 3.14e-09 ***
## enginequad    -0.4512     0.4790  -0.942    0.346
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Threshold coefficients:
##     Estimate Std. Error z value
## 0|1  -1.9344     0.1609 -12.023
## 1|2  -1.0522     0.1269  -8.290
## 2|3  -0.3181     0.1142  -2.786
## 3|4   0.4510     0.1167   3.865
## 4|5   1.5990     0.1525  10.486
```

```
confint(olm) |> print()
```

```
##                  2.5 %     97.5 %
## enginecustom -2.225924 -1.1212219
## enginequad   -1.396963  0.4927549
```

Compared to Bevy, custom engines are much less likely to have missing features.