

Graphic Design with ggplot2

Concepts of the {ggplot2} Package Pt. 1:

Solution Exercise 2

Cédric Scherer // rstudio::conf // July 2022

Exercise 2

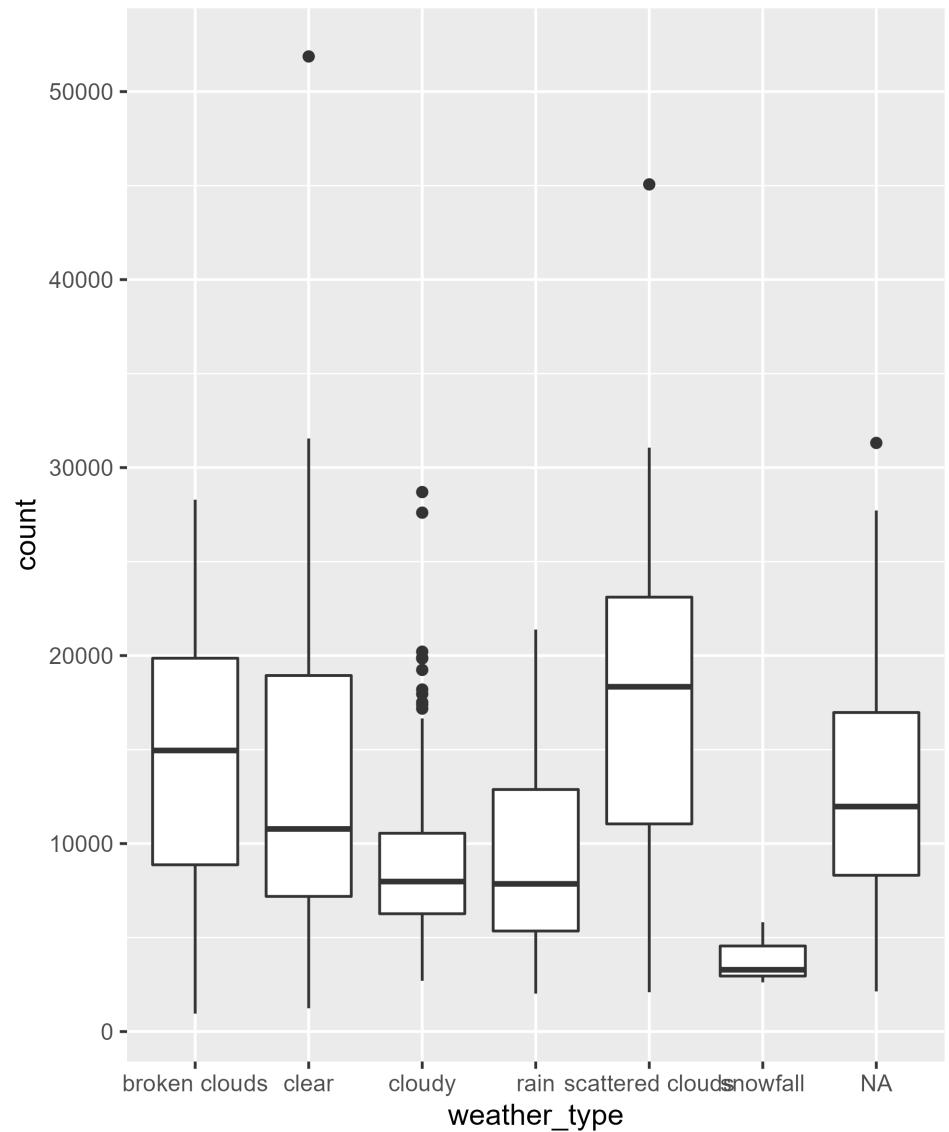
- Explore the TfL bike sharing data visually:
create a boxplot of counts per weather type
 - Turn the plot into a jitter strips plot (random noise across the x axis)
 - Combine both chart types (jittered points on top of the boxplots)
 - Bonus: Sort the boxplot-jitter hybrid by median counts
 - Apply your favorite theme to the plot.
 - Add meaningful labels.
 - Bonus: Explore other chart types to visualize the distributions.
- Save the plot as a vector graphic with a decent plot size.

Import Data (if not yet)

```
1 bikes <- readr::read_csv(  
2   here::here("data", "london-bikes-custom.csv"),  
3   col_types = "Dcffffillllddddc"  
4 )  
5  
6 bikes$season <- forcats::fct_inorder(bikes$season)  
7  
8 library(tidyverse)
```

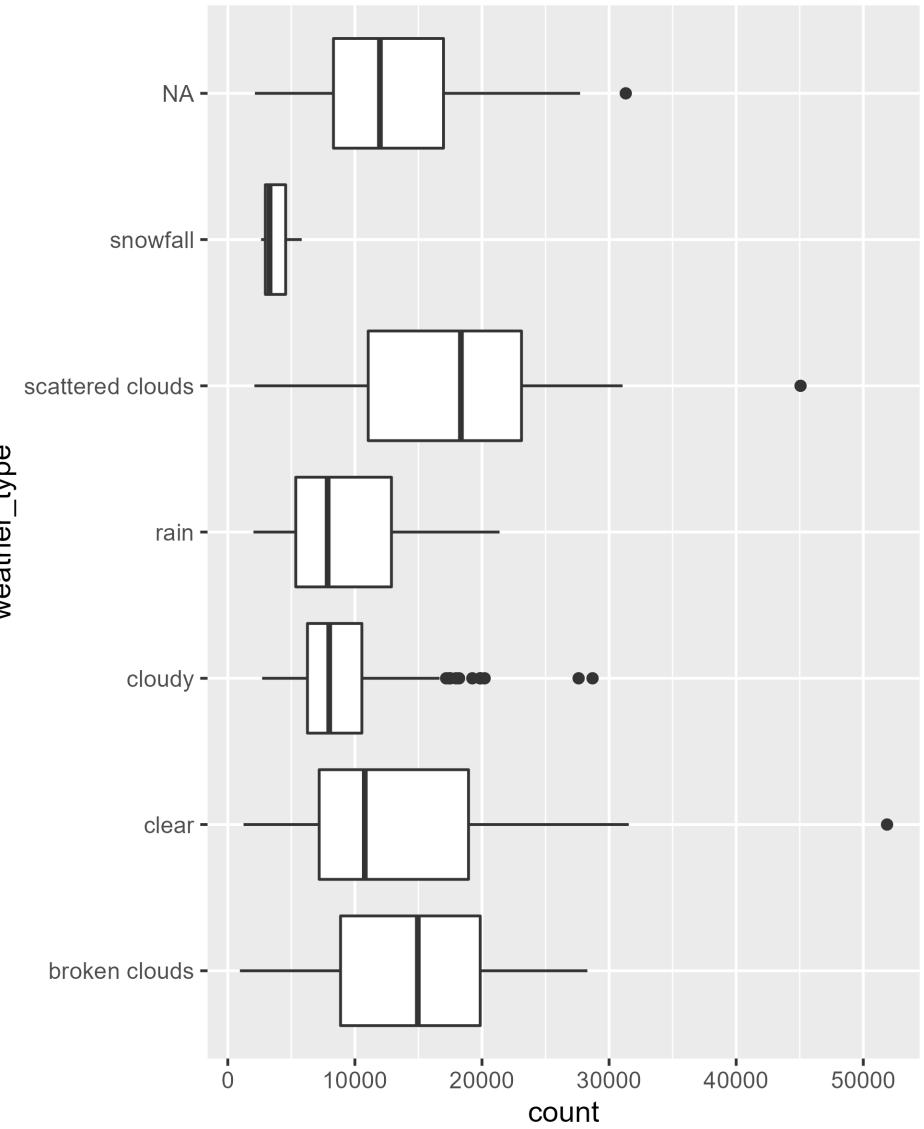
Boxplot of Counts vs. Weather Type

```
1 ggplot(  
2   bikes,  
3   aes(x = weather_type, y = count)  
4 ) +  
5 geom_boxplot()
```



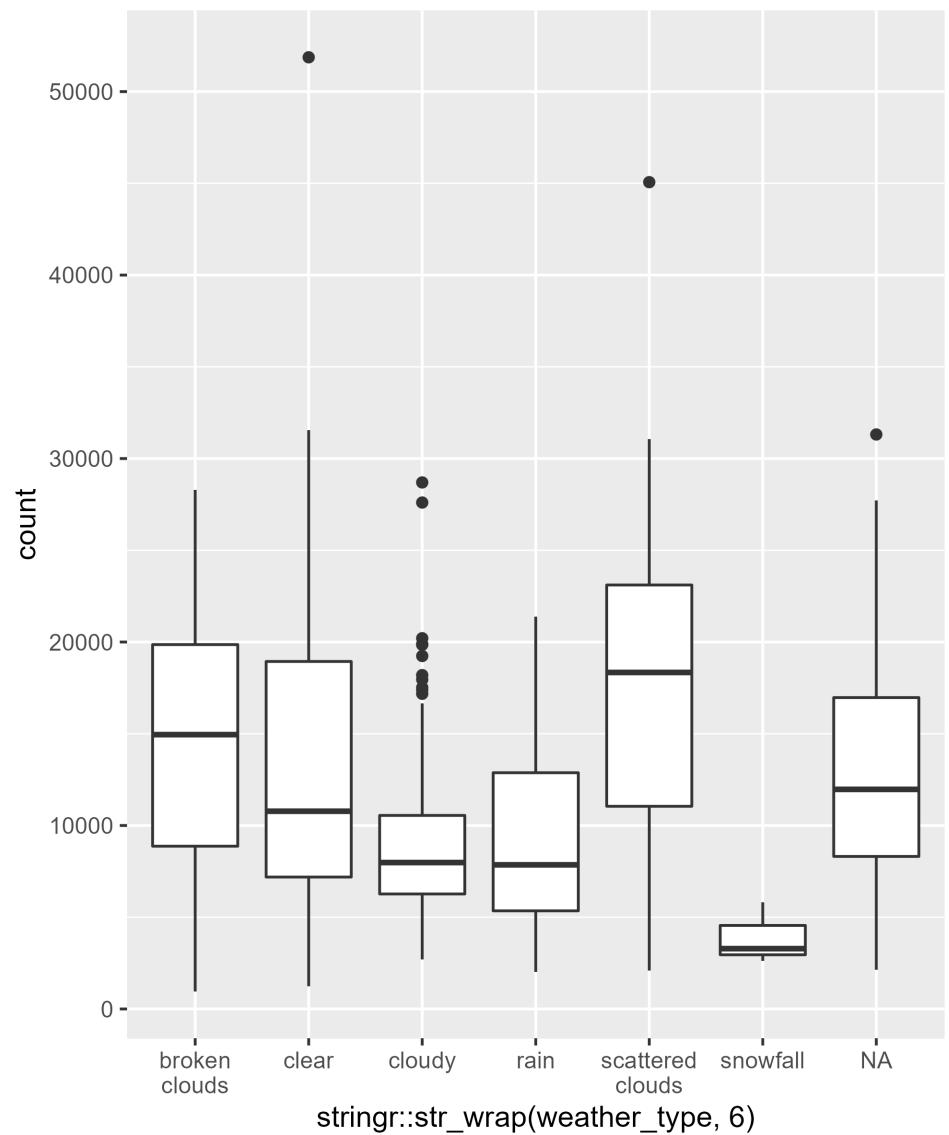
Avoid Overlapping Axis Labels

```
1 ggplot(  
2   bikes,  
3   aes(x = count, y = weather_type))  
4 ) +  
5 geom_boxplot()
```



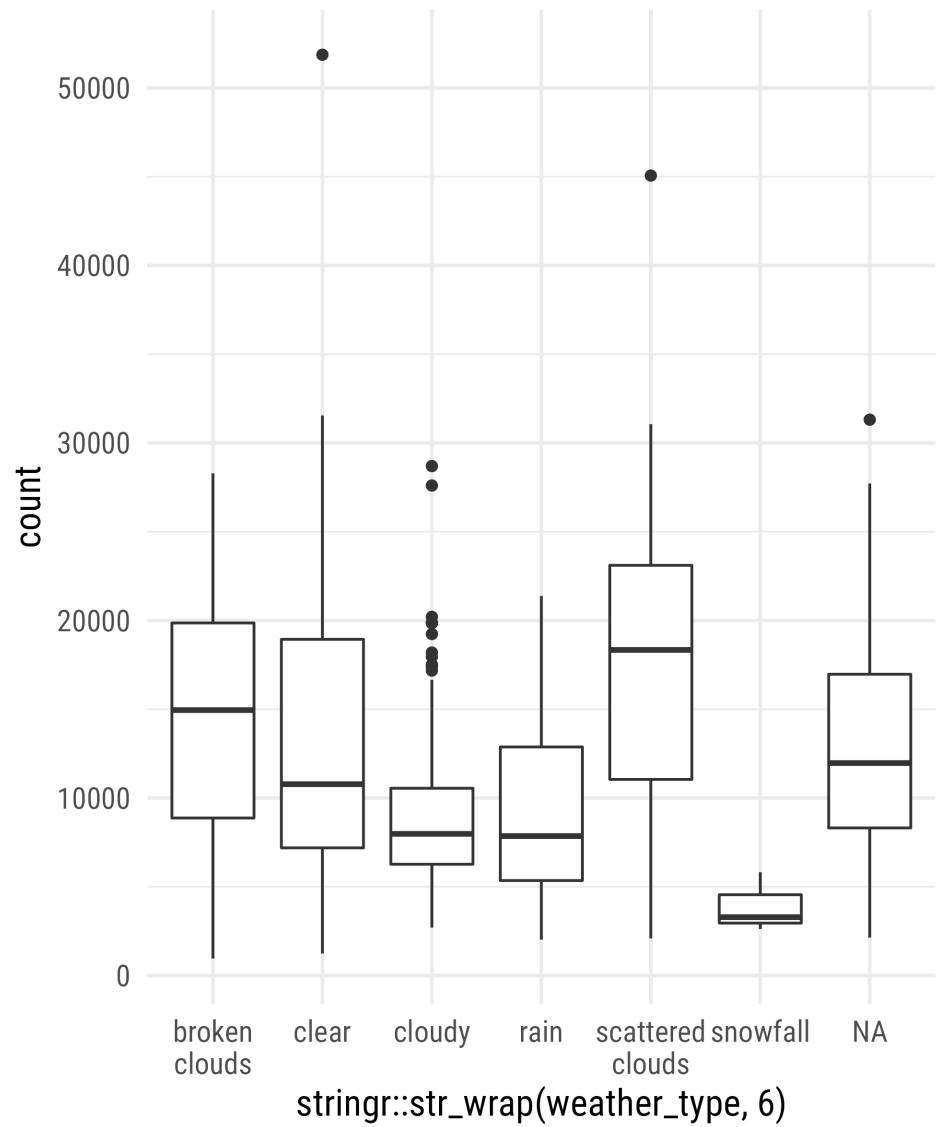
Avoid Overlapping Axis Labels

```
1 ggplot(  
2   bikes,  
3   aes(x = stringr::str_wrap(weather_type, 6),  
4        y = count)  
5 ) +  
6 geom_boxplot()
```



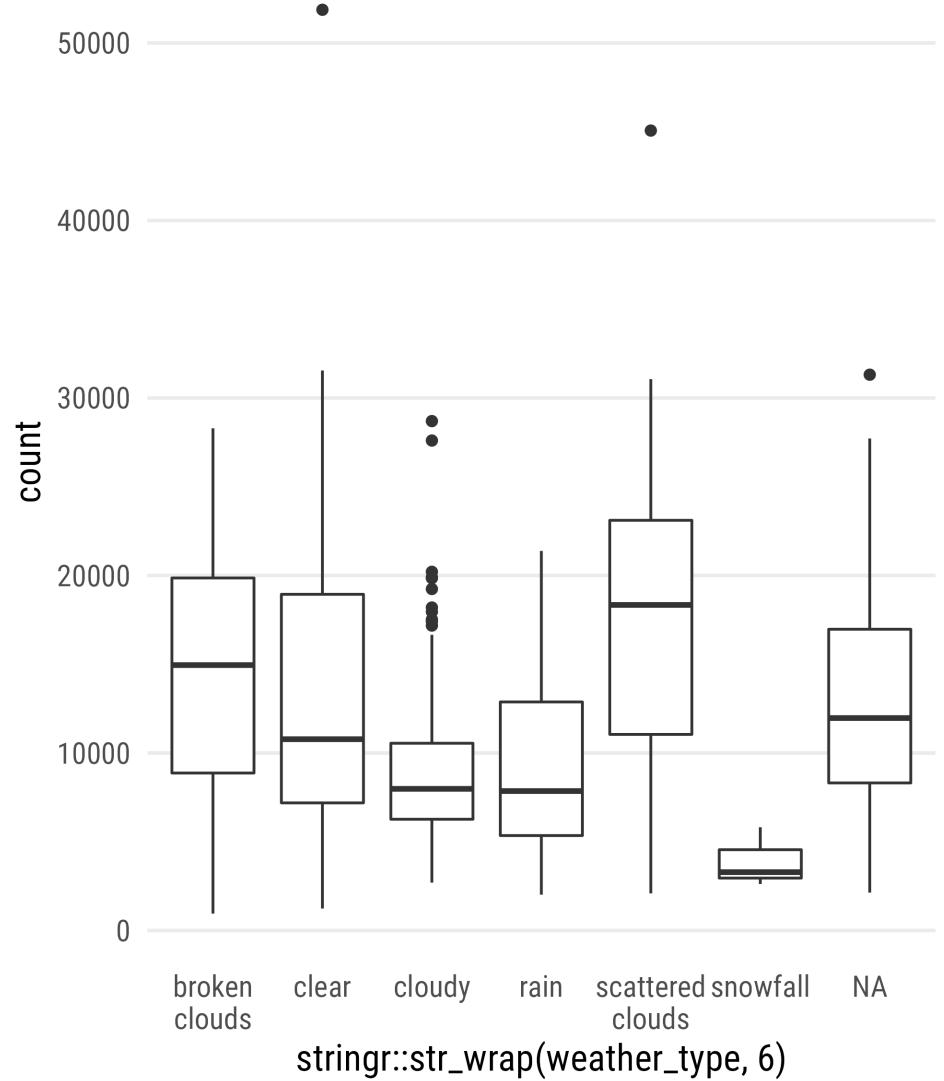
Apply a Theme

```
1 theme_set(theme_minimal(  
2   base_size = 14,  
3   base_family = "Roboto Condensed"  
4 ))  
5  
6 ggplot(  
7   bikes,  
8   aes(x = stringr::str_wrap(weather_type, 6),  
9     y = count)  
10 ) +  
11 geom_boxplot()
```



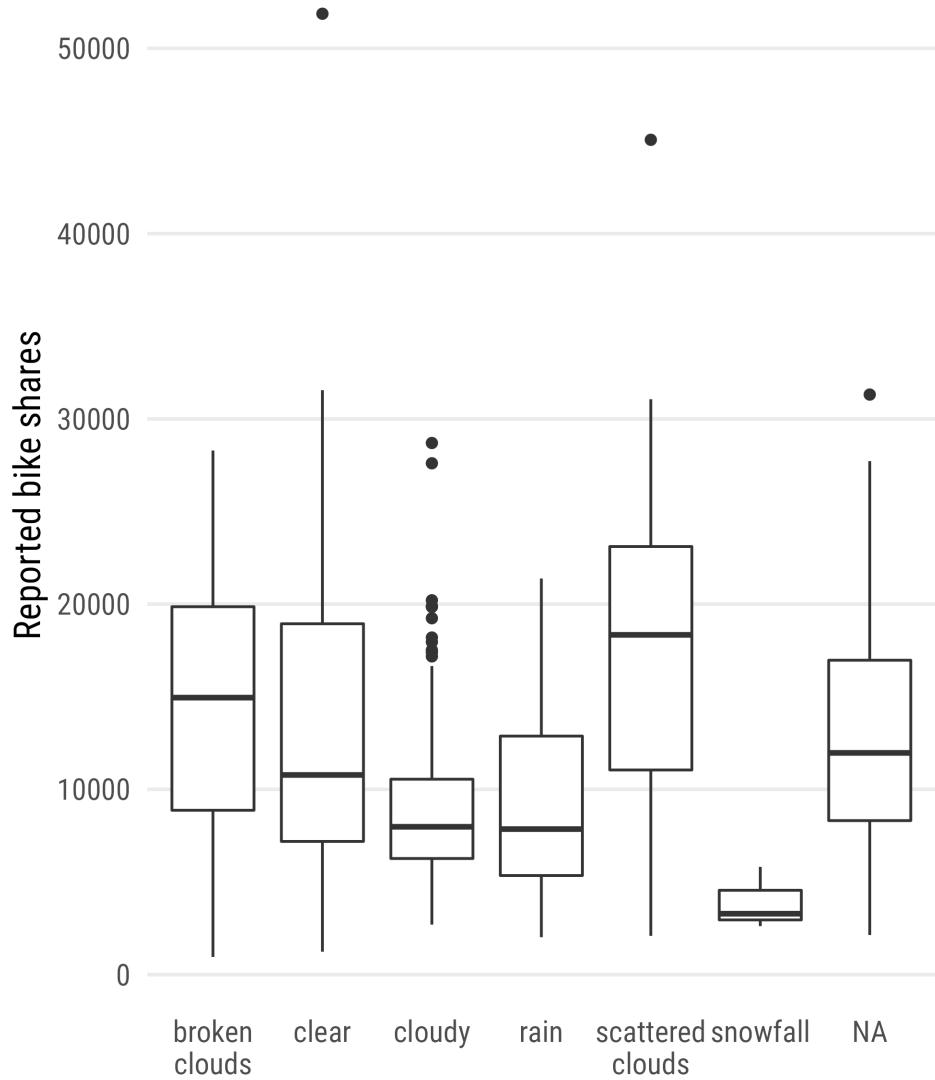
Customize the Theme

```
1 theme_set(theme_minimal(  
2   base_size = 14,  
3   base_family = "Roboto Condensed"  
4 ))  
5  
6 theme_update(  
7   panel.grid.major.x = element_blank(),  
8   panel.grid.minor = element_blank()  
9 )  
10 ggplot(  
11   bikes,  
12   aes(x = stringr::str_wrap(weather_type, 6),  
13         y = count)  
14 ) +  
15 geom_boxplot()
```



Add Meaningful Labels

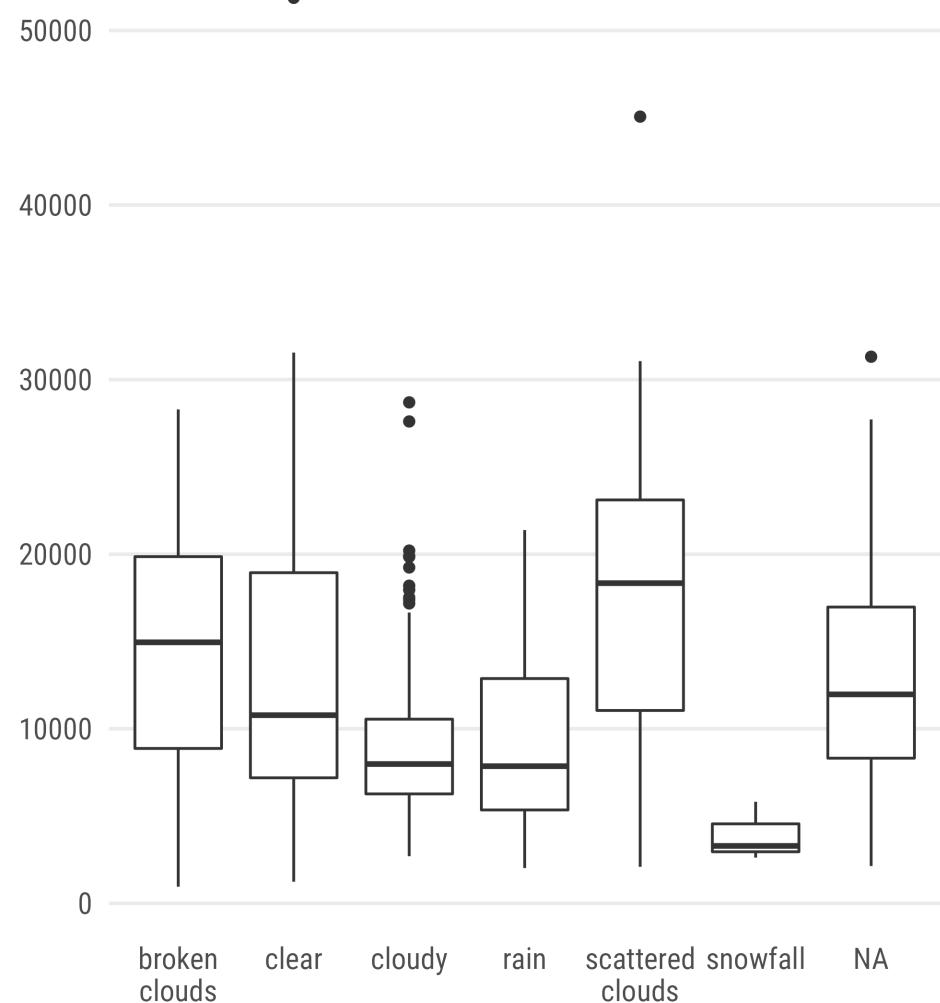
```
1 ggplot(  
2   bikes,  
3   aes(x = stringr::str_wrap(weather_type, 6),  
4        y = count)  
5 ) +  
6 geom_boxplot() +  
7 labs(x = NULL, y = "Reported bike shares")
```



Add Meaningful Labels

```
1 ggplot(  
2   bikes,  
3   aes(x = stringr::str_wrap(weather_type, 6),  
4       y = count)  
5 ) +  
6 geom_boxplot() +  
7 labs(  
8   x = NULL, y = NULL,  
9   title = "Reported bike shares by weather type")  
10 )
```

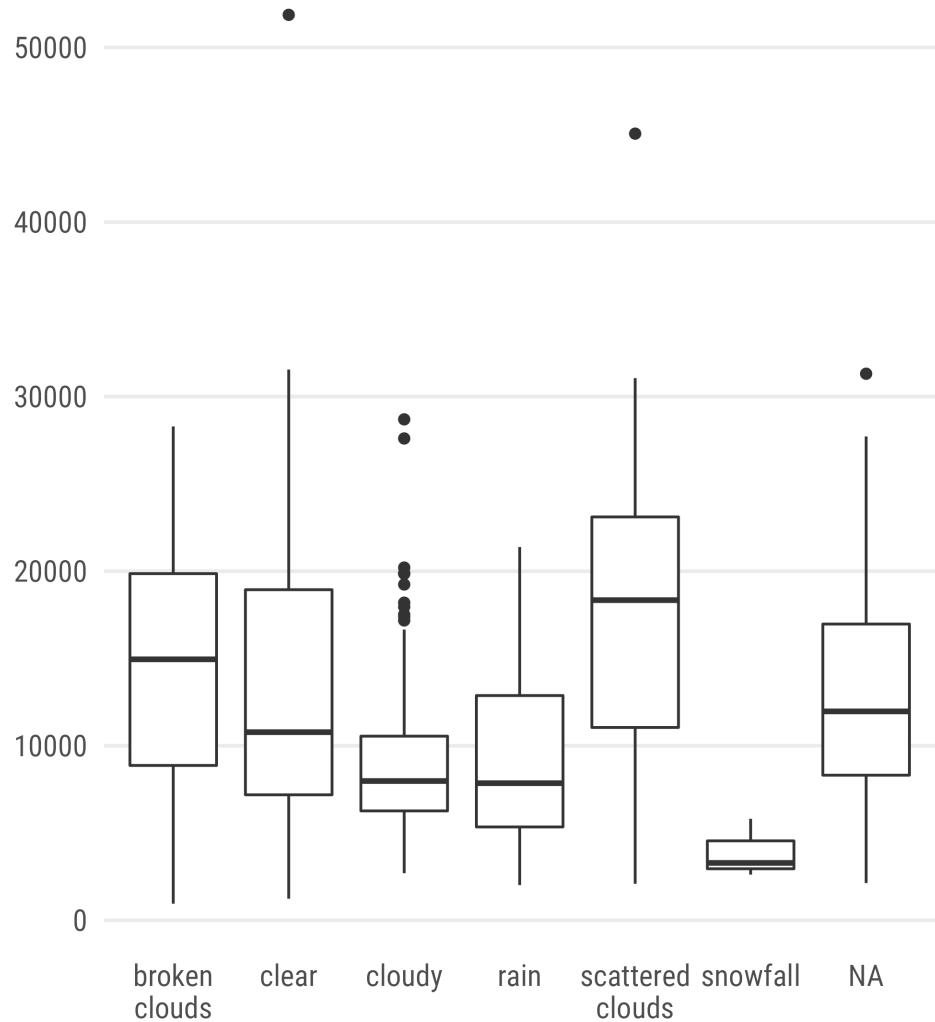
Reported bike shares by weather type



Add Meaningful Labels

```
1 theme_update(  
2   plot.title.position = "plot",  
3   axis.title = element_blank()  
4 )  
5  
6 ggplot(  
7   bikes,  
8   aes(x = stringr::str_wrap(weather_type, 6),  
9     y = count)  
10 ) +  
11 geom_boxplot() +  
12 ggttitle("Reported bike shares by weather type")
```

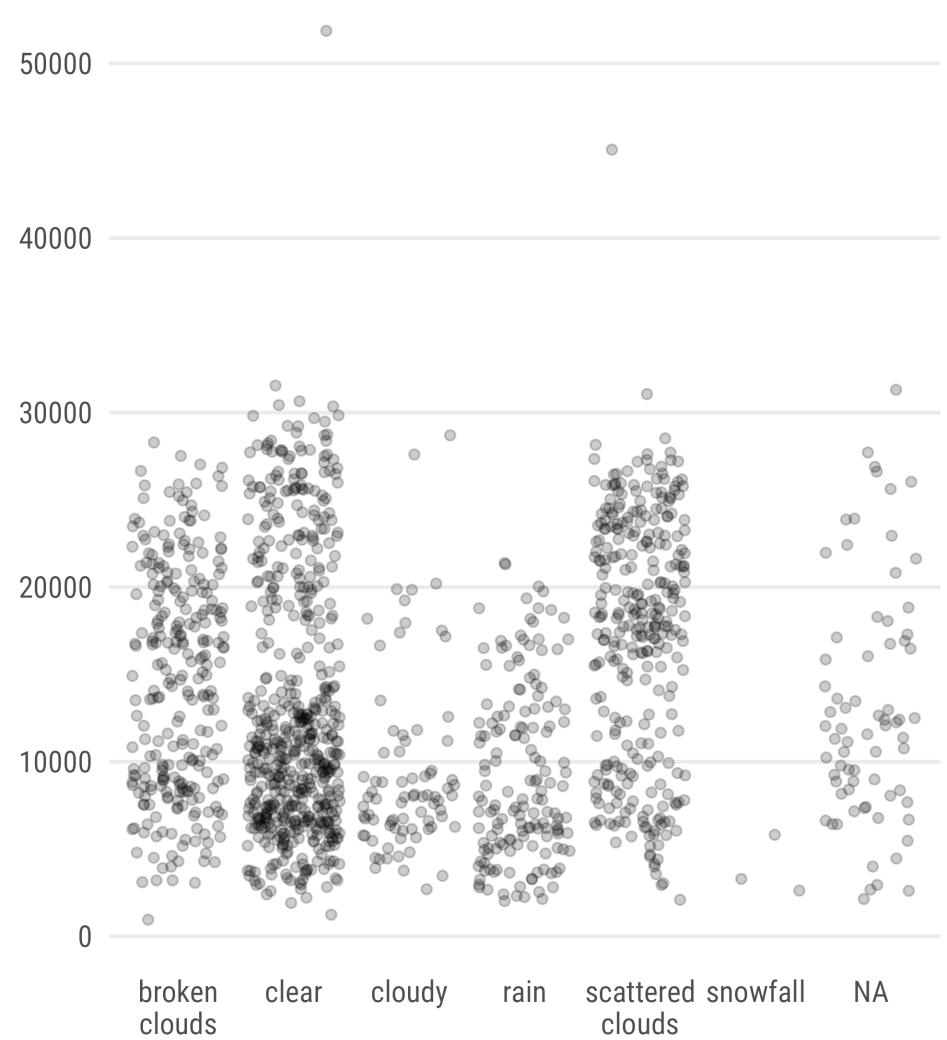
Reported bike shares by weather type



Jitter Strips of Counts per Weather Type

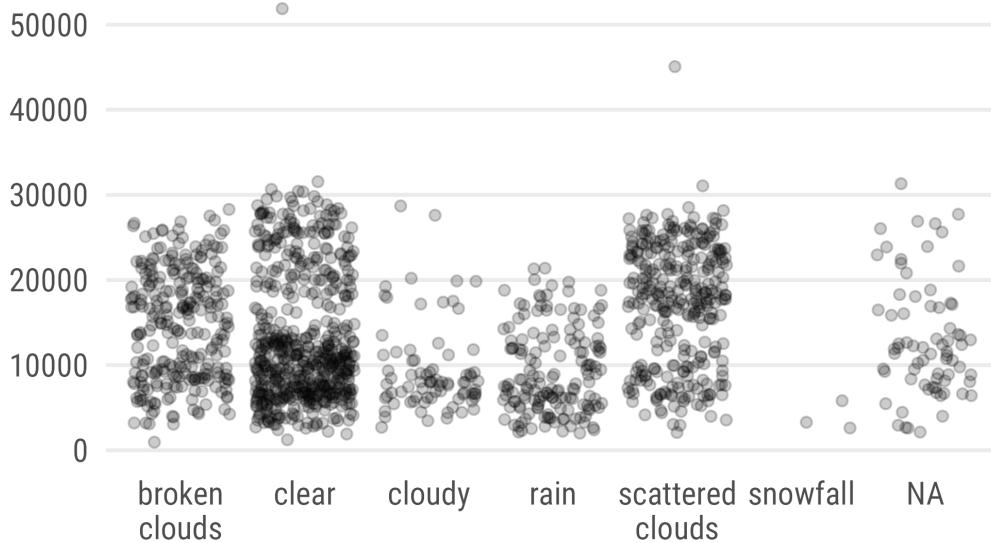
```
1 ggplot(  
2   bikes,  
3   aes(x = stringr::str_wrap(weather_type, 6),  
4       y = count)  
5 ) +  
6   geom_jitter(  
7     alpha = .2  
8 ) +  
9   ggtitle("Reported bike shares by weather type")
```

Reported bike shares by weather type

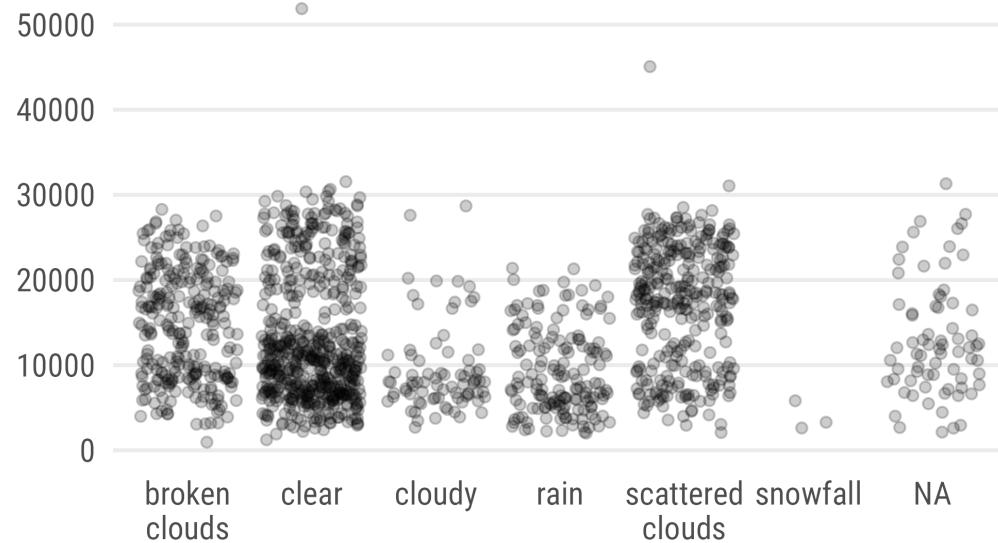


Jitter Strips of Counts per Weather Type

```
1 ggplot(  
2   bikes,  
3   aes(x = str_wrap(weather_type, 6),  
4       y = count)  
5 ) +  
6 geom_point(  
7   position = "jitter",  
8   alpha = .2  
9 )
```



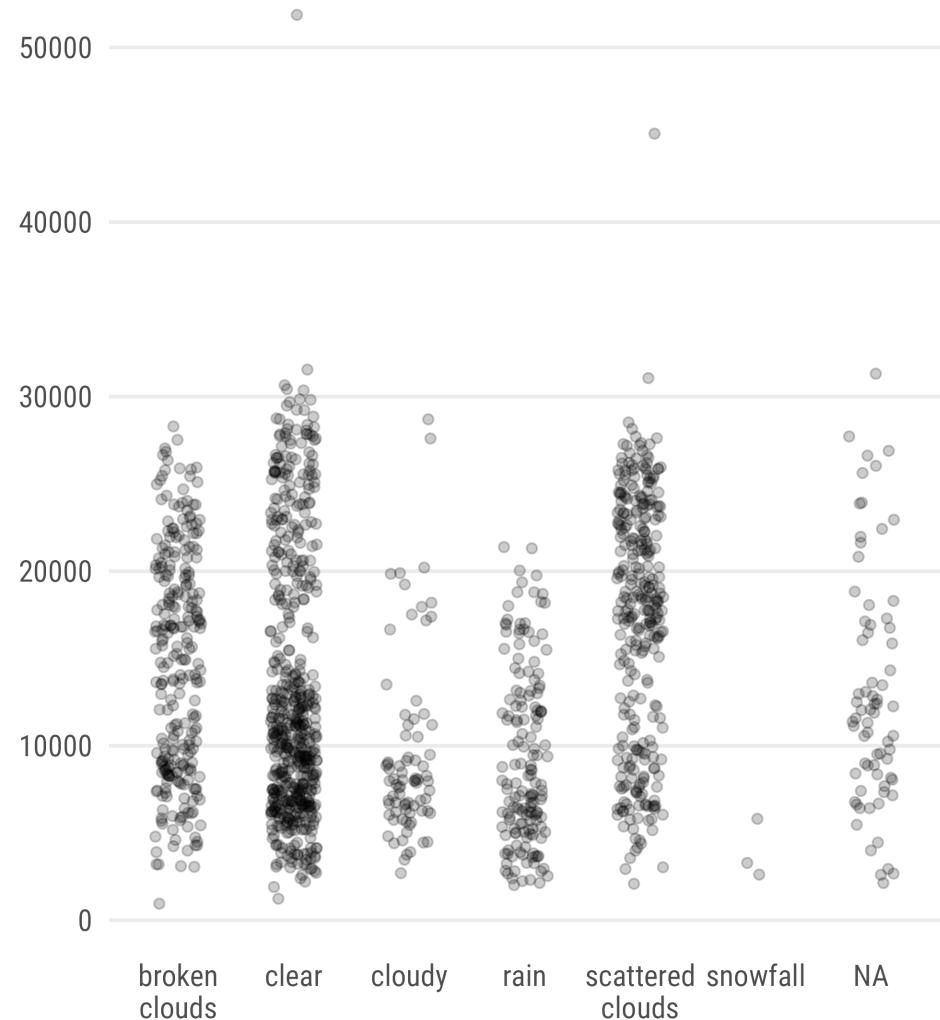
```
1 ggplot(  
2   bikes,  
3   aes(x = str_wrap(weather_type, 6),  
4       y = count)  
5 ) +  
6 geom_point(  
7   position = position_jitter(),  
8   alpha = .2  
9 )
```



Jitter Strips of Counts vs. Weather Type

```
1 ggplot(  
2   bikes,  
3   aes(x = str_wrap(weather_type, 6),  
4       y = count)  
5 ) +  
6 geom_point(  
7   position = position_jitter(  
8     seed = 2022,  
9     width = .2,  
10    height = 0  
11 ),  
12   alpha = .2  
13 ) +  
14 ggttitle("Reported bike shares by weather type")
```

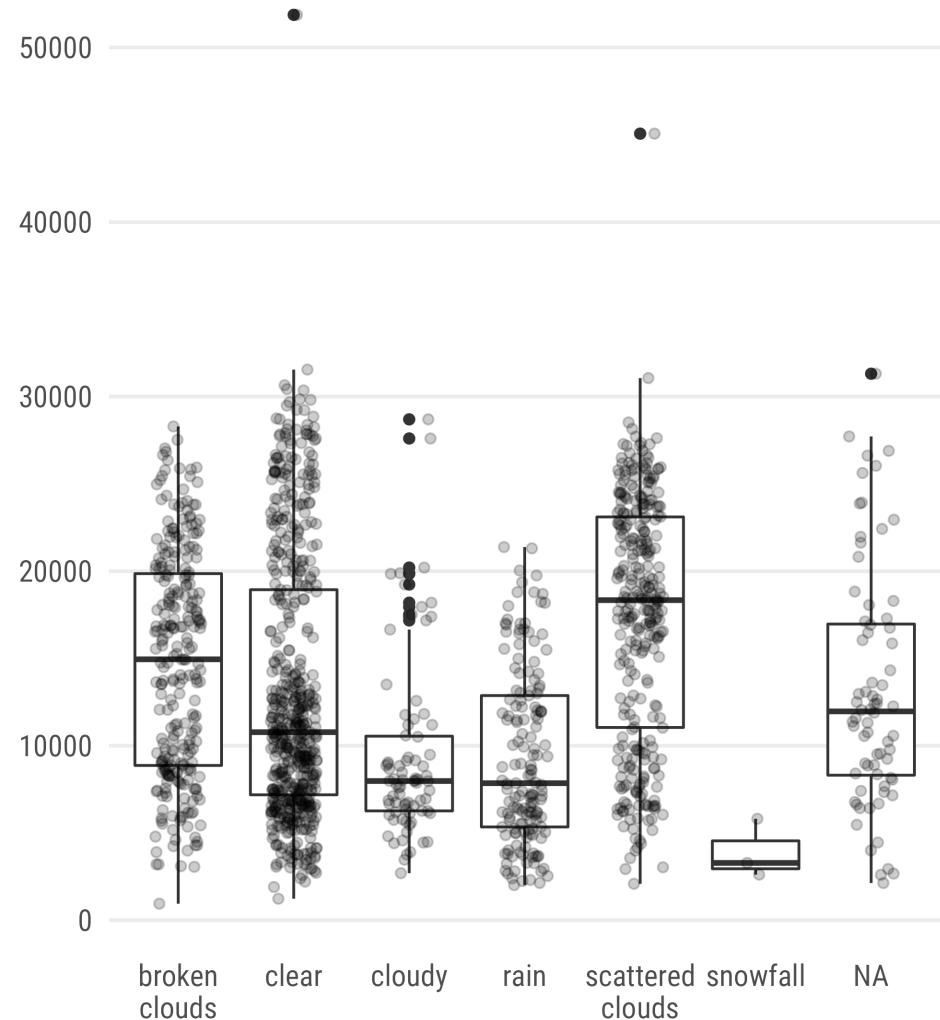
Reported bike shares by weather type



Boxplot + Jitter Strip Hybrid

```
1 ggplot(  
2   bikes,  
3   aes(x = str_wrap(weather_type, 6),  
4       y = count)  
5 ) +  
6 geom_boxplot() +  
7 geom_point(  
8   position = position_jitter(  
9     seed = 2022,  
10    width = .2,  
11    height = 0  
12 ),  
13 alpha = .2  
14 ) +  
15 ggttitle("Reported bike shares by weather type")
```

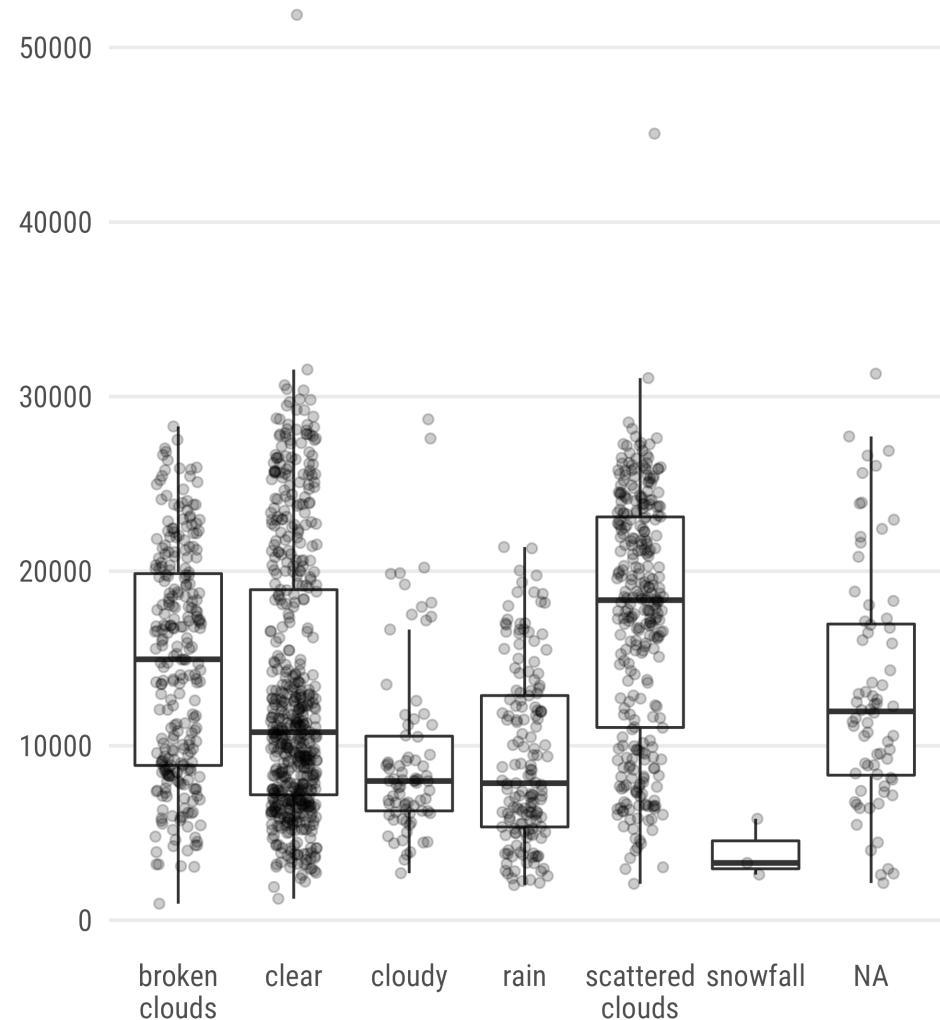
Reported bike shares by weather type



Boxplot + Jitter Strip Hybrid

```
1 ggplot(  
2   bikes,  
3   aes(x = str_wrap(weather_type, 6),  
4       y = count)  
5 ) +  
6   geom_boxplot(  
7     outlier.shape = NA  
8     # outlier.color = "transparent"  
9     # outlier.alpha = 0  
10 ) +  
11  geom_point(  
12    position = position_jitter(  
13      seed = 2022,  
14      width = .2,  
15      height = 0  
16    ),  
17    alpha = .2  
18 ) +  
19  ggttitle("Reported bike shares by weather type")
```

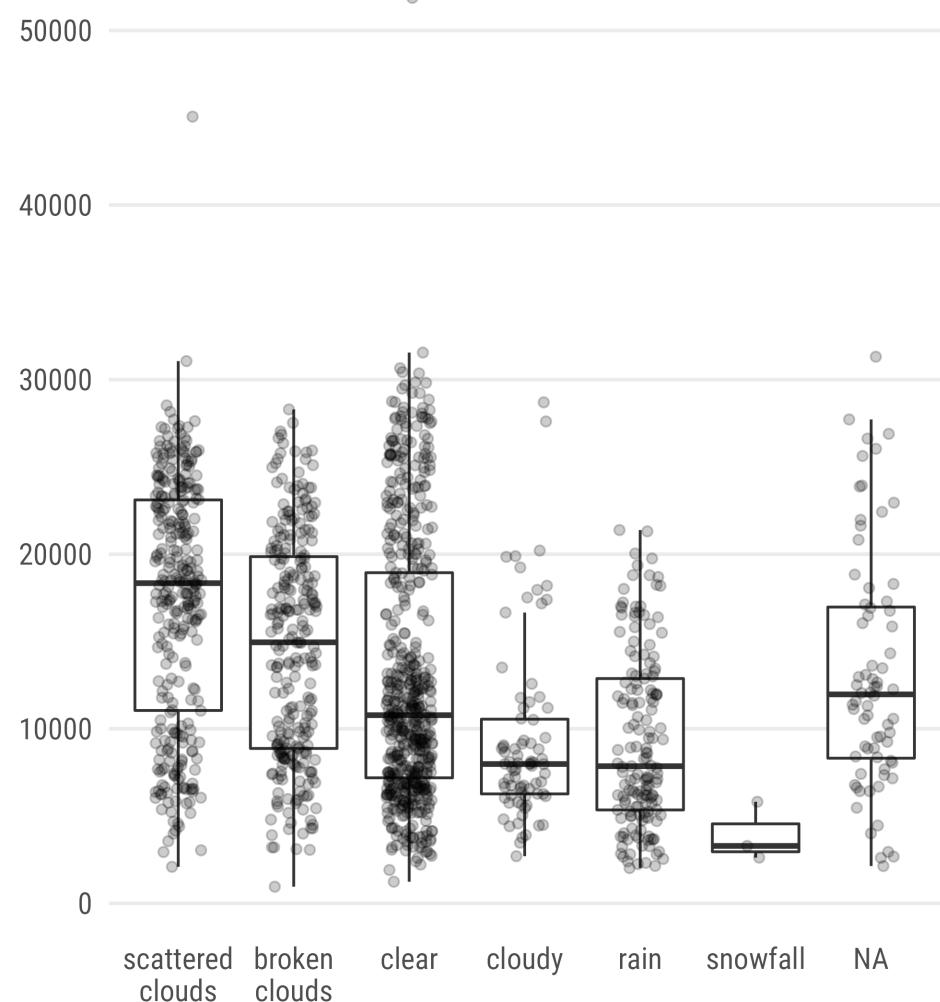
Reported bike shares by weather type



Bonus: Sort Weather Types

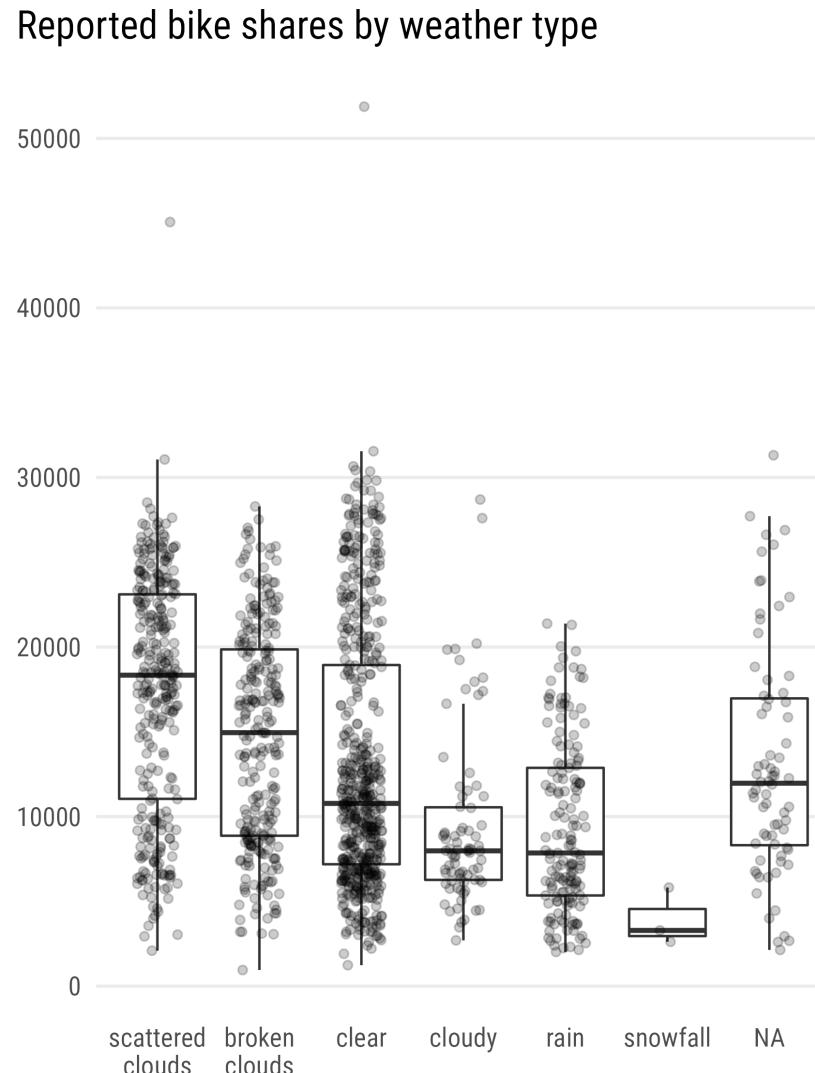
```
1 ggplot(
2   bikes,
3   aes(
4     x = forcats::fct_reorder(
5       str_wrap(weather_type, 6), -count
6     ),
7     y = count)
8 ) +
9   geom_boxplot(
10   outlier.shape = NA
11   # outlier.color = "transparent"
12   # outlier.alpha = 0
13 ) +
14   geom_point(
15   position = position_jitter(
16     seed = 2022,
17     width = .2,
18     height = 0
19   ),
20   alpha = .2
21 ) +
22   ggtitle("Reported bike shares by weather type")
```

Reported bike shares by weather type



Save the Plot

```
1 ggsave(here:::here("exercises", "plots", "02_concepts_pt1_ex2.pdf"),  
2       width = 5, height = 6.5, device = cairo_pdf)
```



Alternative Chart Types to Visualize Distributions

Let's Update our Data Set

```
1 bikes <-  
2   bikes %>%  
3   mutate(  
4     weather_type_fct = stringr::str_wrap(  
5       weather_type, 6  
6     ),  
7     weather_type_fct =forcats::fct_reorder(  
8       weather_type_fct, -count  
9     )  
10    )  
11  
12 levels(bikes$weather_type_fct)  
  
[1] "scattered\nclouds"  "broken\nclouds"      "clear"  
[4] "cloudy"             "rain"              "snowfall"
```

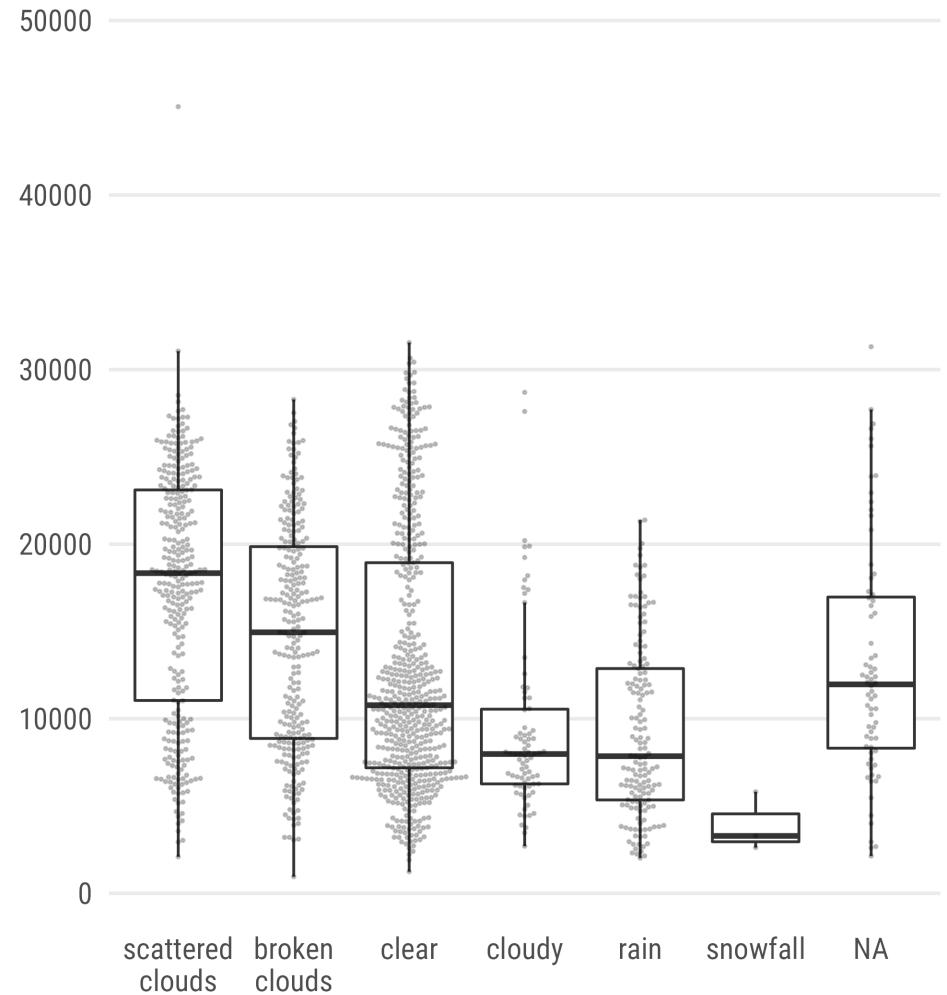
Let's Store Our ggplot Setup

```
1 g <-
2   ggplot(
3     bikes,
4     aes(x = weather_type_fct,
5          y = count)
6   ) +
7   ggtitle(
8     "Reported bike shares by weather type"
9   )
```

Beeswarm Plots with `{ggbeeswarm}`

```
1 g +  
2   geom_boxplot(  
3     outlier.shape = NA  
4   ) +  
5   ggbeeswarm::geom_beeswarm(  
6     size = .3,  
7     alpha = .2,  
8     cex = .6  
9   )
```

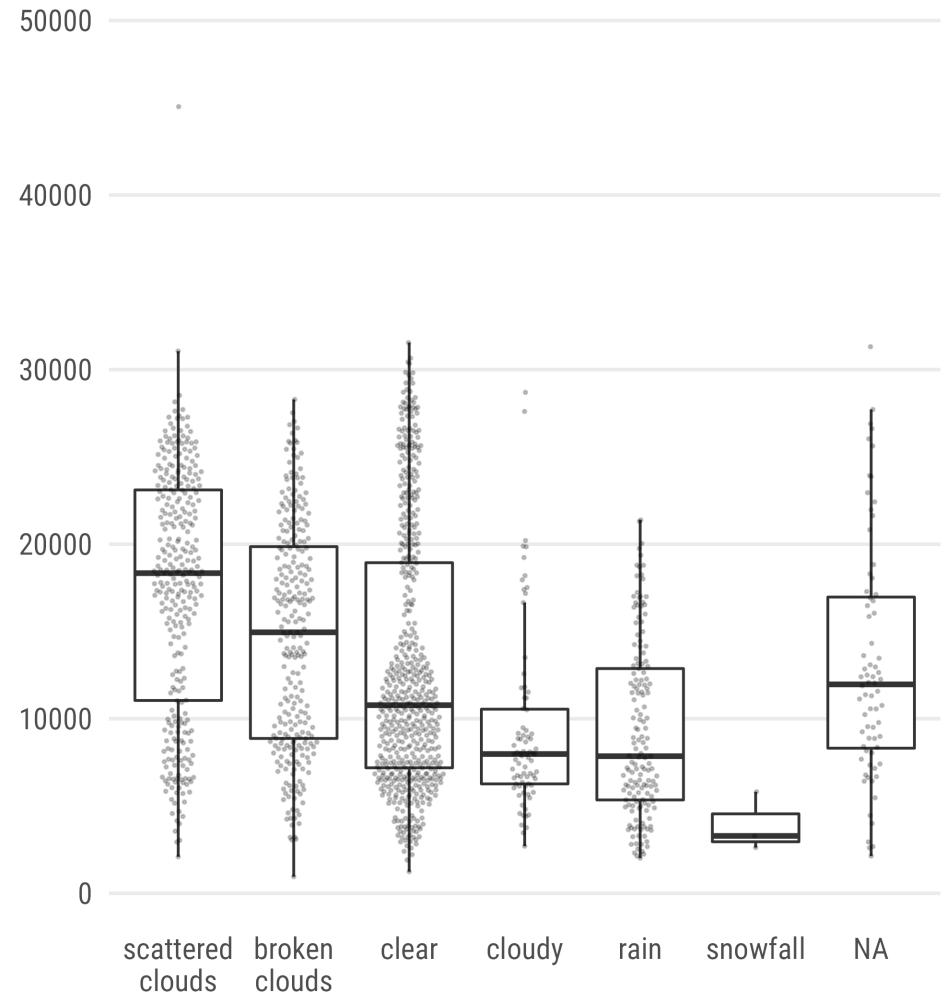
Reported bike shares by weather type



Beeswarm Plots with `{ggbeeswarm}`

```
1 g +  
2   geom_boxplot(  
3     outlier.shape = NA  
4   ) +  
5   ggbeeswarm::geom_quasirandom(  
6     size = .3,  
7     alpha = .2,  
8     width = .3,  
9     varwidth = TRUE  
10 )
```

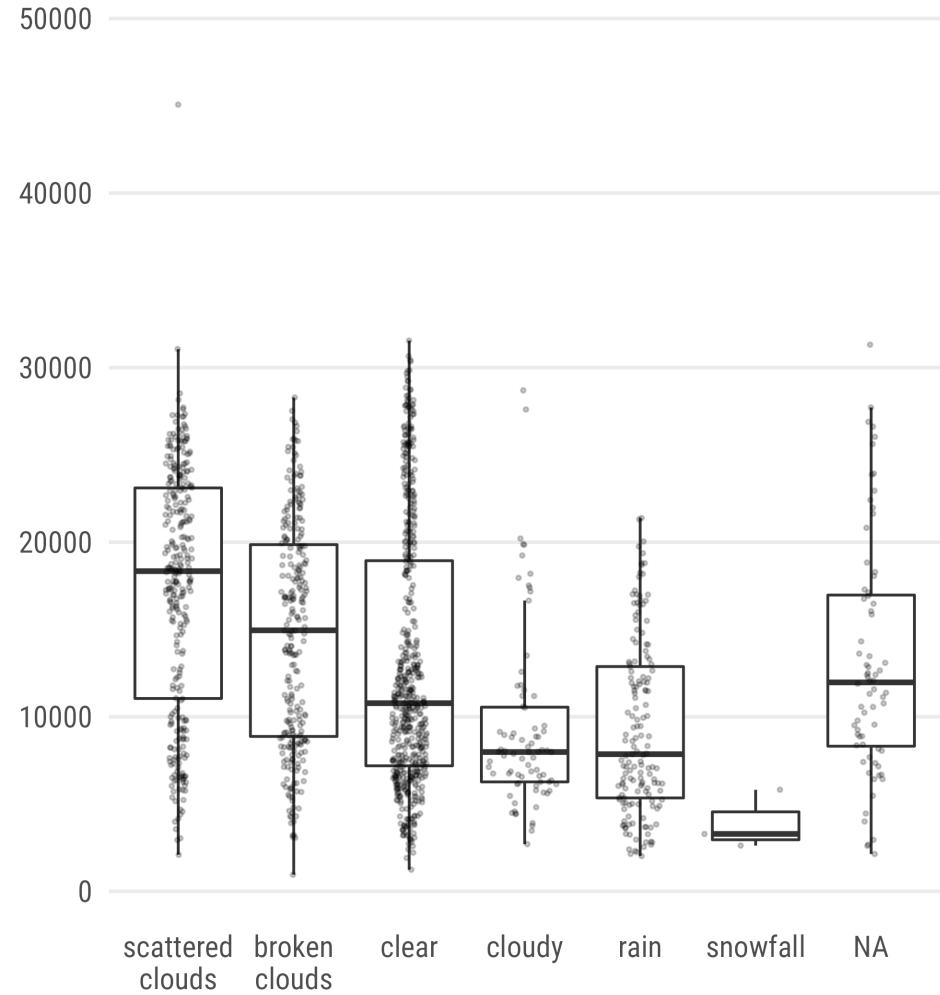
Reported bike shares by weather type



Sina Plots with {ggforce}

```
1 g +  
2   geom_boxplot(  
3     outlier.shape = NA  
4   ) +  
5   ggforce::geom_sina(  
6     size = .5,  
7     alpha = .2,  
8     maxwidth = 1.2  
9   )
```

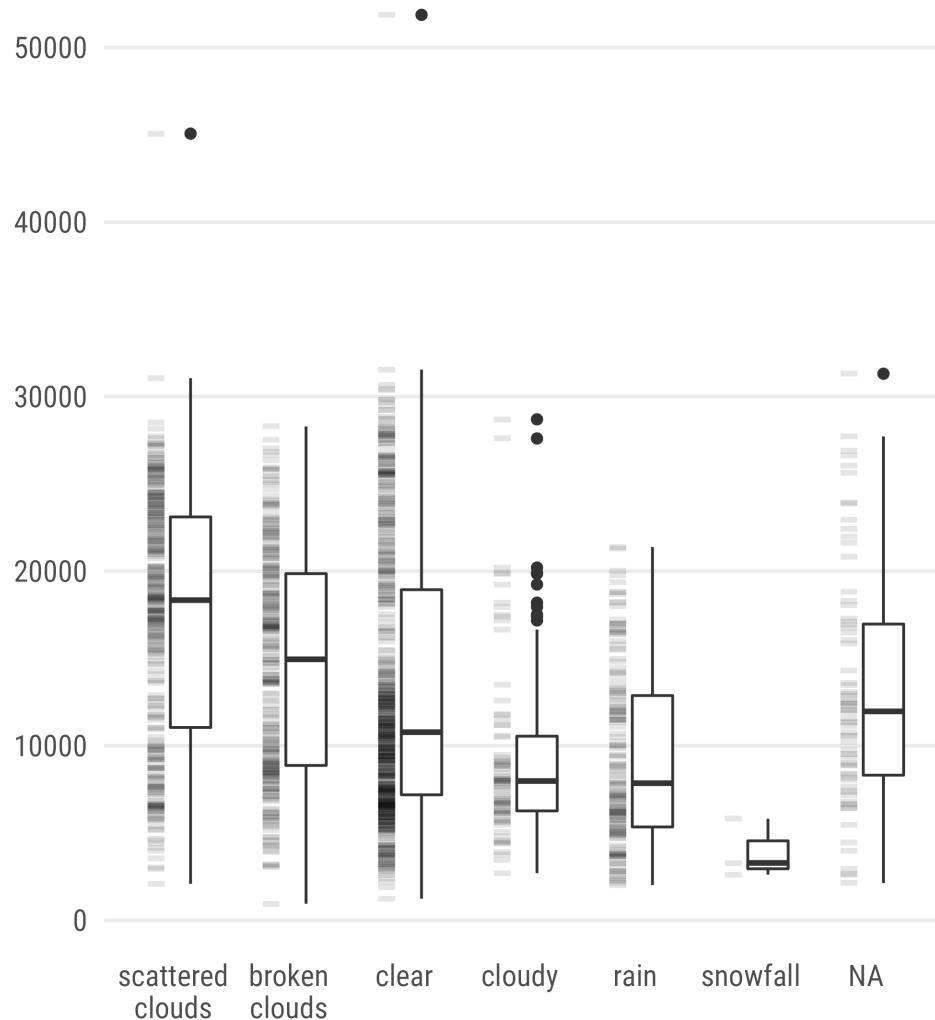
Reported bike shares by weather type



Barcode Strips

```
1 g +
2   geom_boxplot(
3     position = position_nudge(x = .15),
4     width = .35
5   ) +
6   geom_point(
7     shape = "-",
8     size = 8,
9     alpha = .1,
10    position = position_nudge(x = -.15)
11  )
```

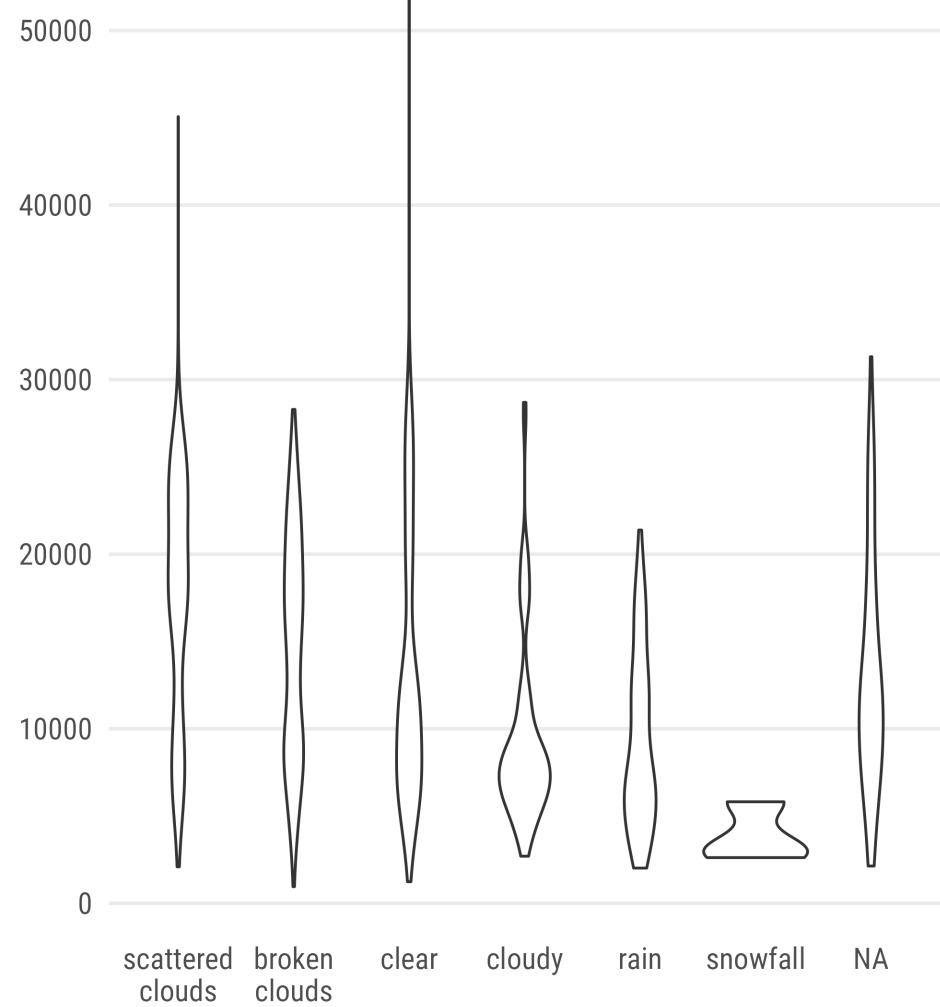
Reported bike shares by weather type



Violin Plots

```
1 g +  
2 geom_violin()
```

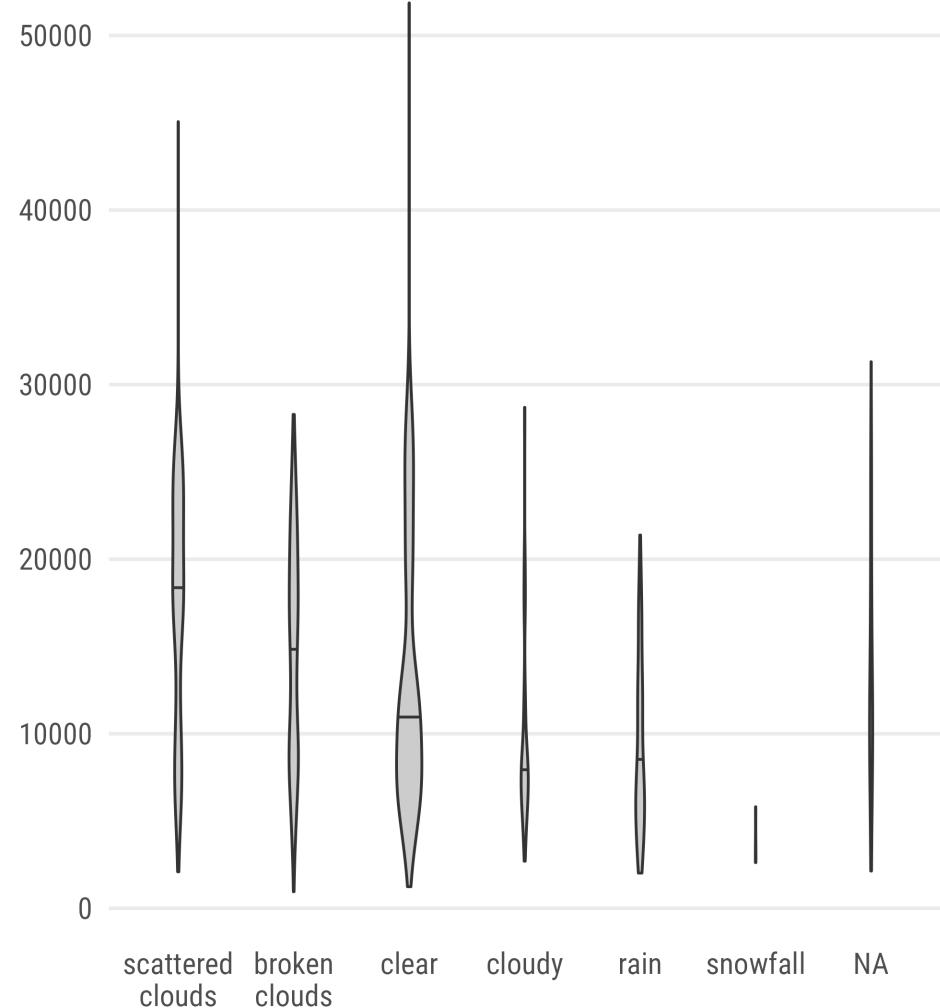
Reported bike shares by weather type



Violin Plots

```
1 g +  
2   geom_violin(  
3     scale = "count",  
4     draw_quantiles = c(.5),  
5     fill = "grey80"  
6   )
```

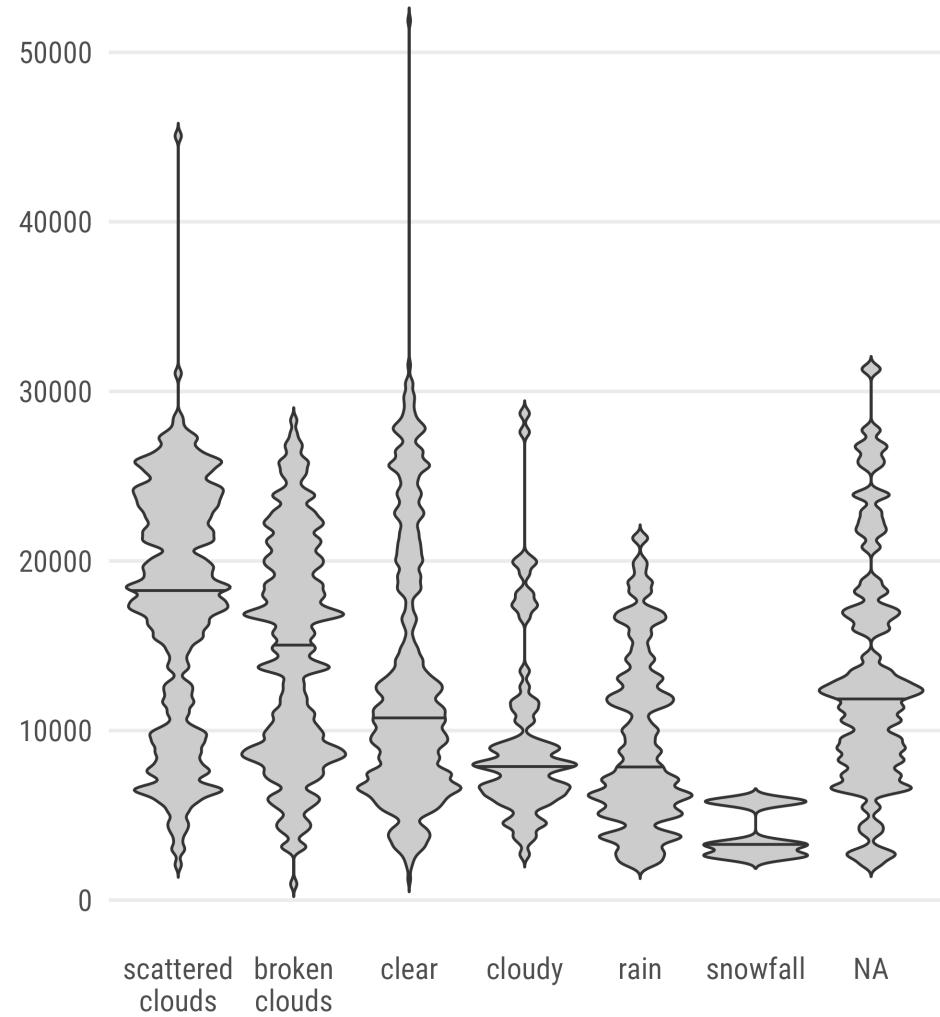
Reported bike shares by weather type



Violin Plots

```
1 g +  
2   geom_violin(  
3     scale = "width",  
4     draw_quantiles = c(.5),  
5     trim = FALSE,  
6     bw = 250,  
7     fill = "grey80"  
8 )
```

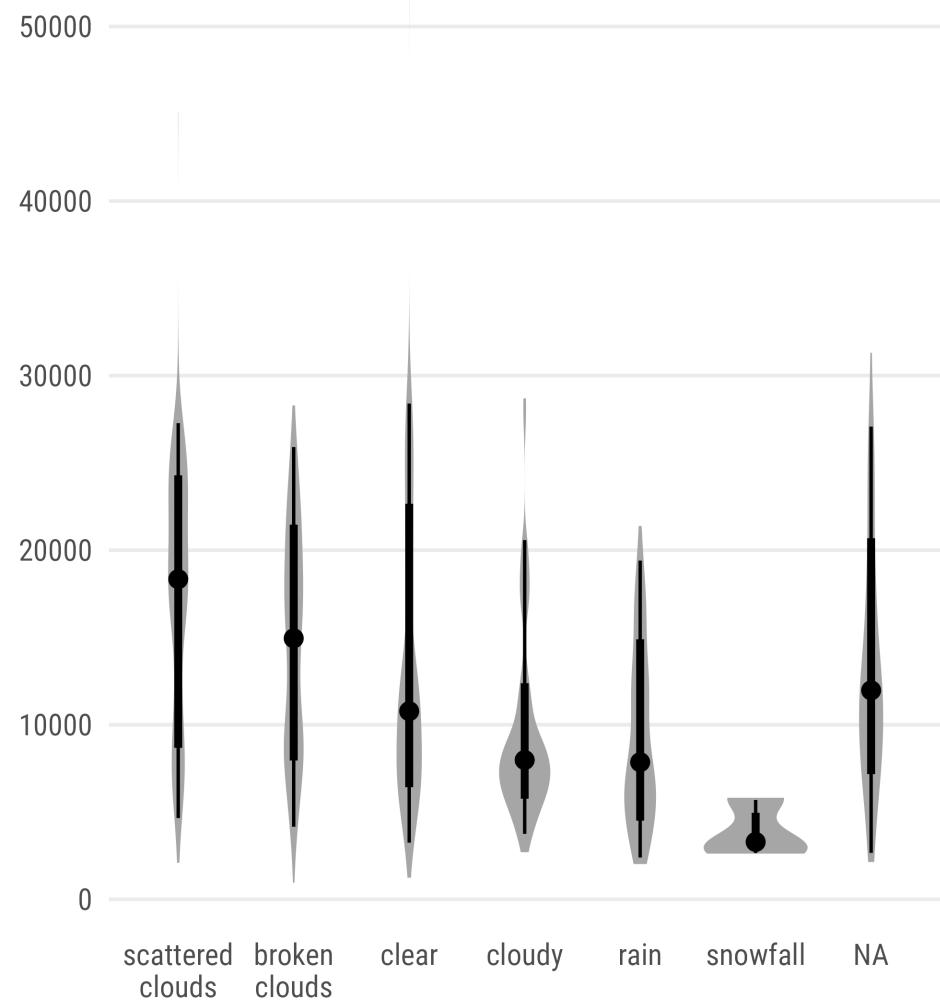
Reported bike shares by weather type



Violin Plots with {ggdist}

```
1 g +  
2 ggdist::stat_eye()
```

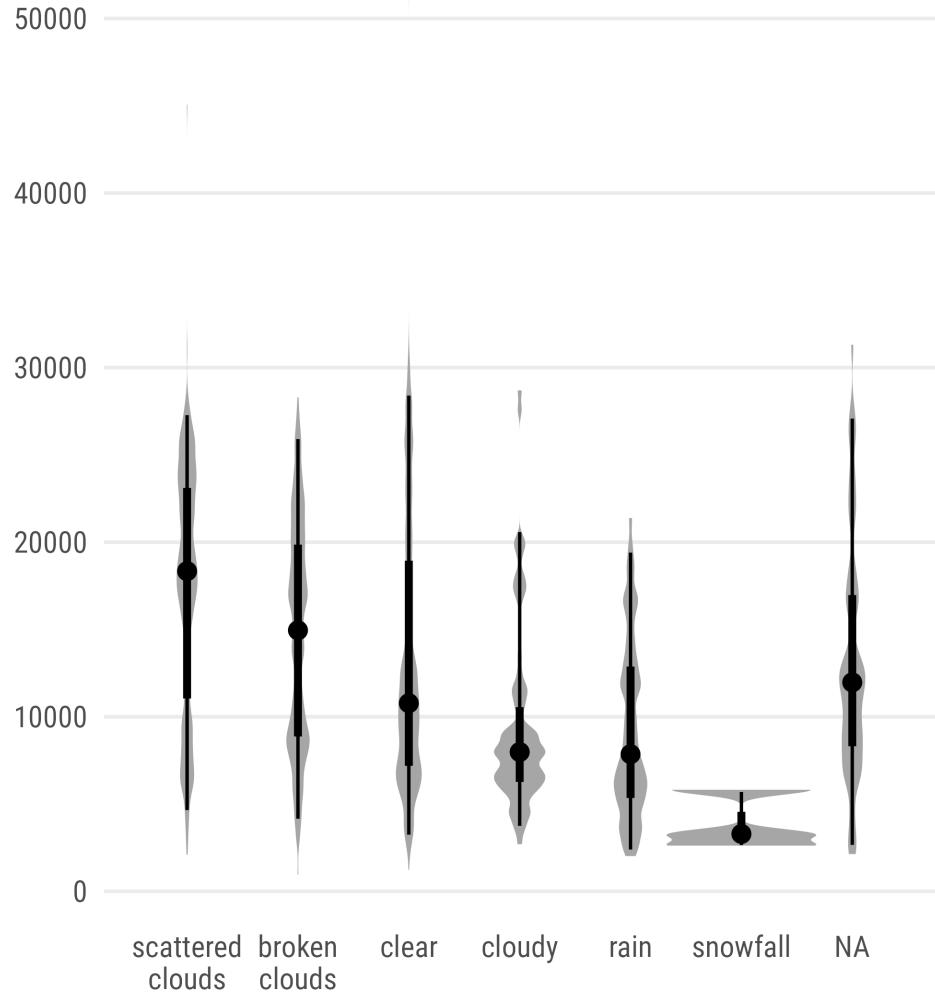
Reported bike shares by weather type



Violin Plots with {ggdist}

```
1 g +  
2   ggdist::stat_eye(  
3     .width = c(0.5, 0.95),  
4     width = 1.5,  
5     adjust = .33  
6   )
```

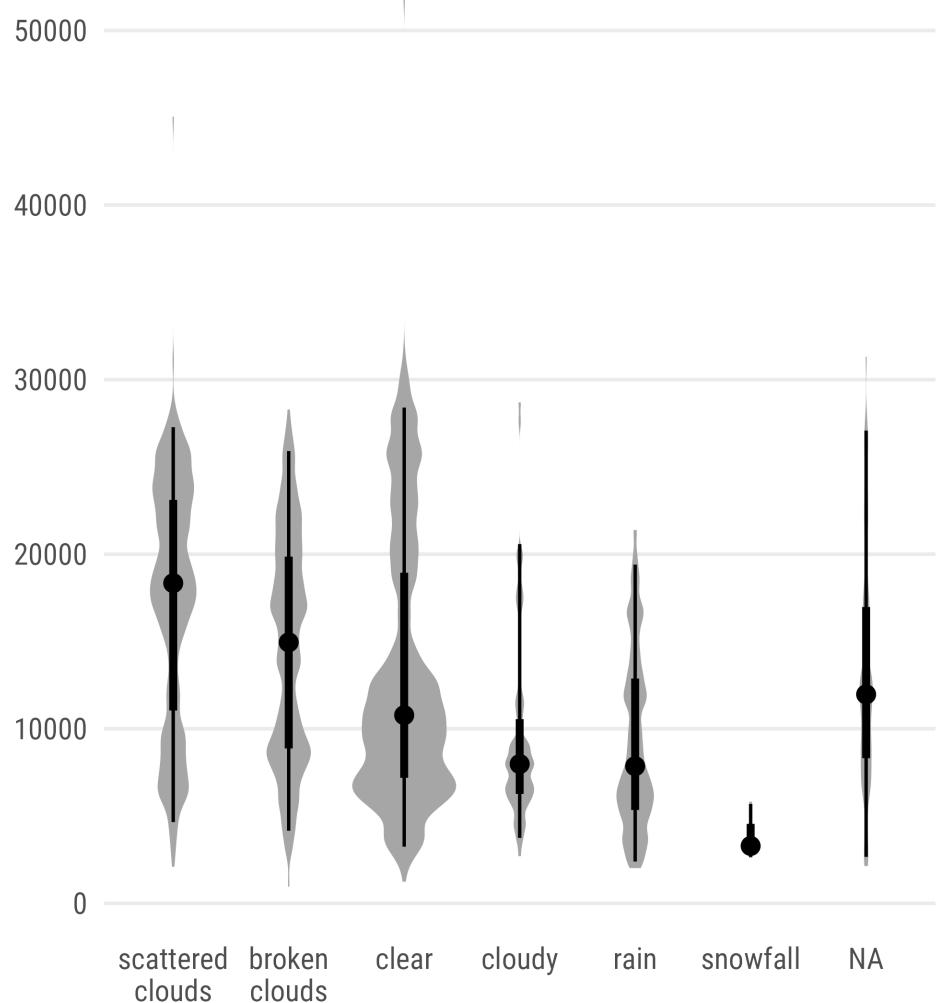
Reported bike shares by weather type



Violin Plots with {ggdist}

```
1 g +  
2   ggdist::stat_eyes(  
3     aes(thickness = stat(f*n)),  
4     .width = c(0.5, 0.95),  
5     adjust = .33  
6   )
```

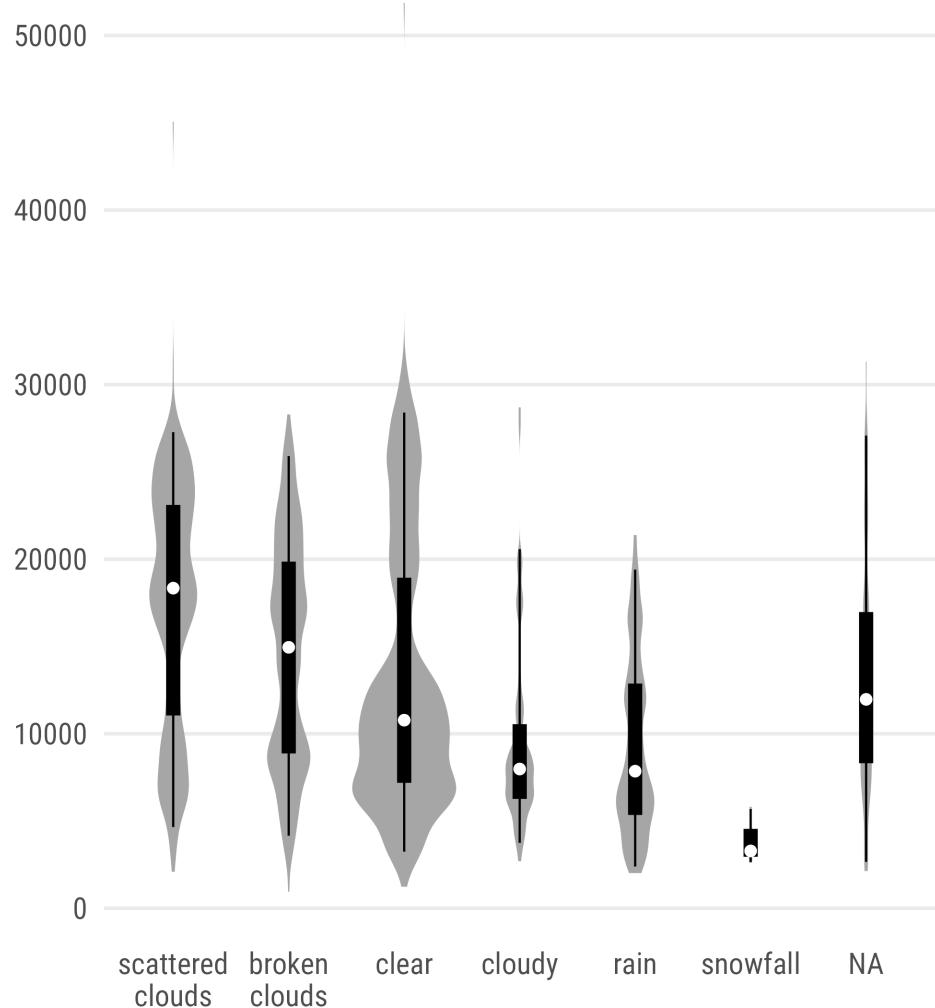
Reported bike shares by weather type



Violin Plots with {ggdist}

```
1 g +  
2   ggdist::stat_eye(  
3     aes(thickness = stat(f*n)),  
4     .width = c(0.5, 0.95),  
5     adjust = .5,  
6     interval_size_range = c(.4, 2.5),  
7     point_size = 1.2,  
8     point_color = "white"  
9 )
```

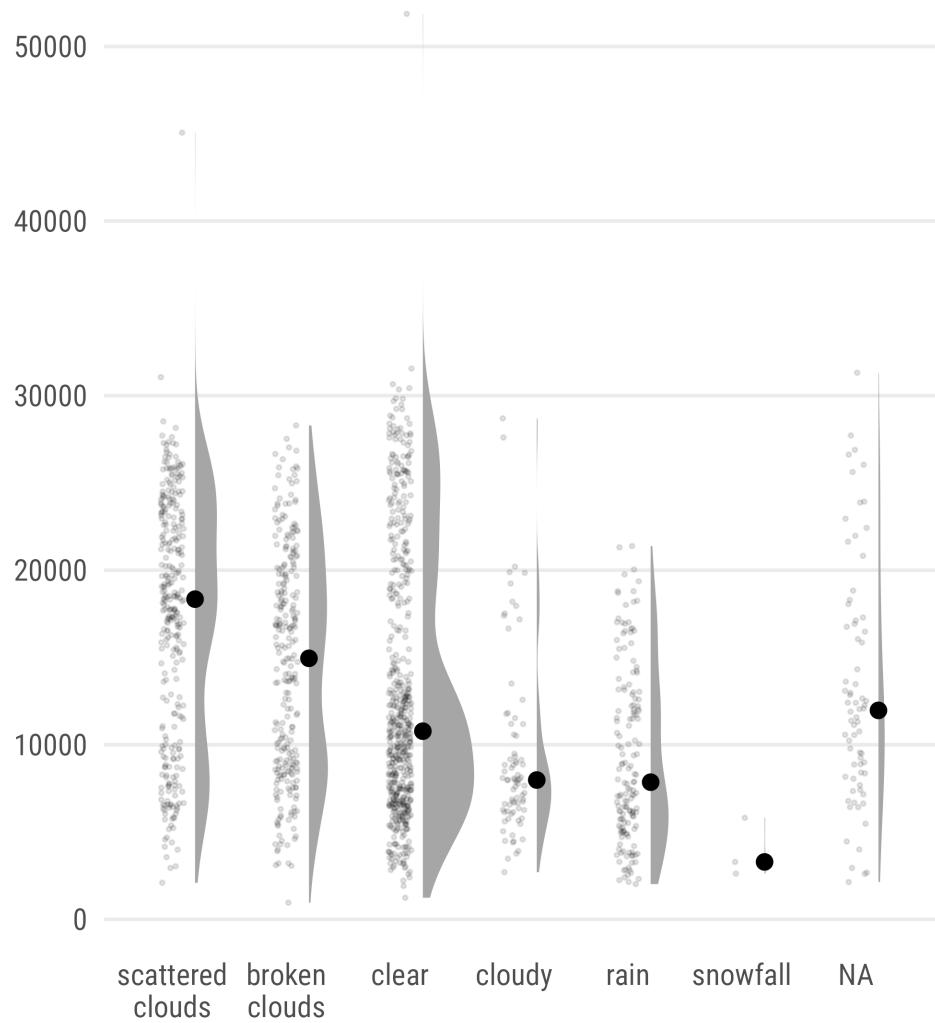
Reported bike shares by weather type



Raincloud Plots with {ggdist}

```
1 g +  
2   ggdist::stat_halfeye(  
3     aes(thickness = stat(f*n)),  
4     .width = 0,  
5     width = .5,  
6     position = position_nudge(x = .2)  
7   ) +  
8   geom_jitter(  
9     width = .1,  
10    size = .5,  
11    alpha = .1  
12  )
```

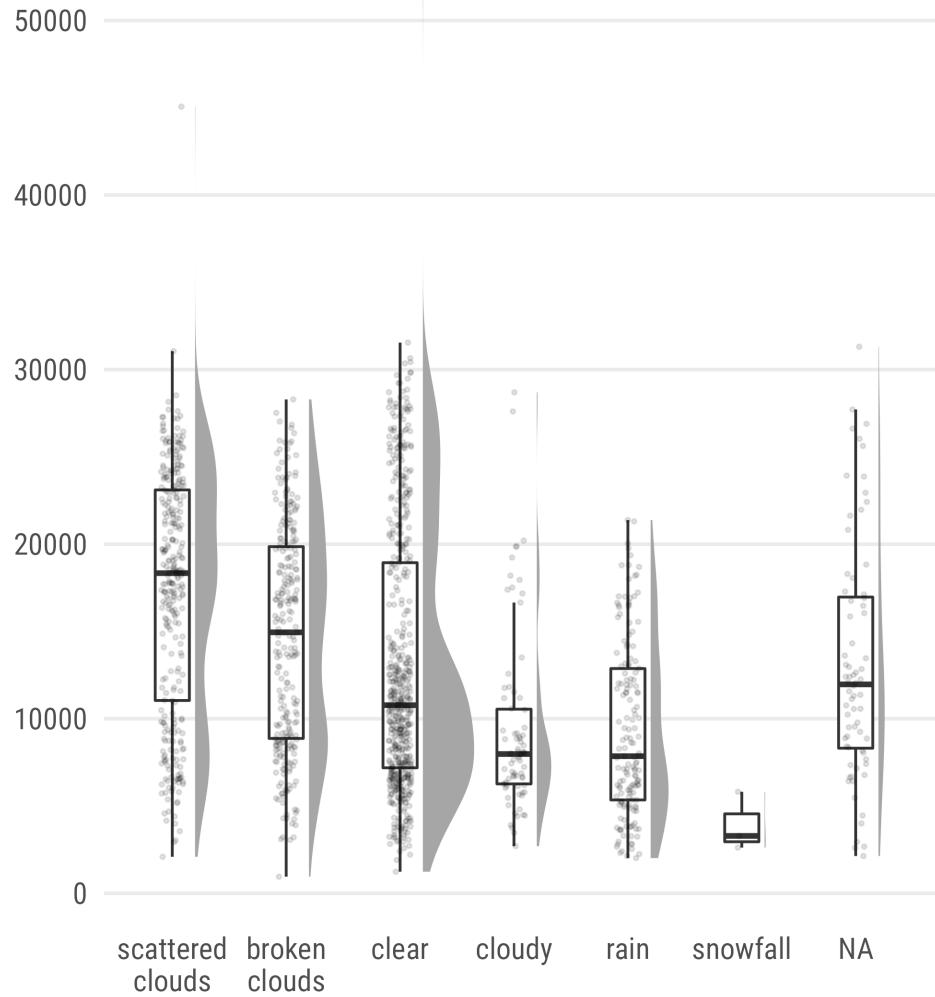
Reported bike shares by weather type



Raincloud Plots with {ggdist}

```
1 g +  
2   ggdist::stat_halfeye(  
3     aes(thickness = stat(f*n)),  
4     color = NA,  
5     width = .5,  
6     position = position_nudge(x = .2)  
7   ) +  
8   geom_boxplot(  
9     width = .3,  
10    outlier.color = NA  
11  ) +  
12  geom_jitter(  
13    width = .1,  
14    size = .5,  
15    alpha = .1  
16  )
```

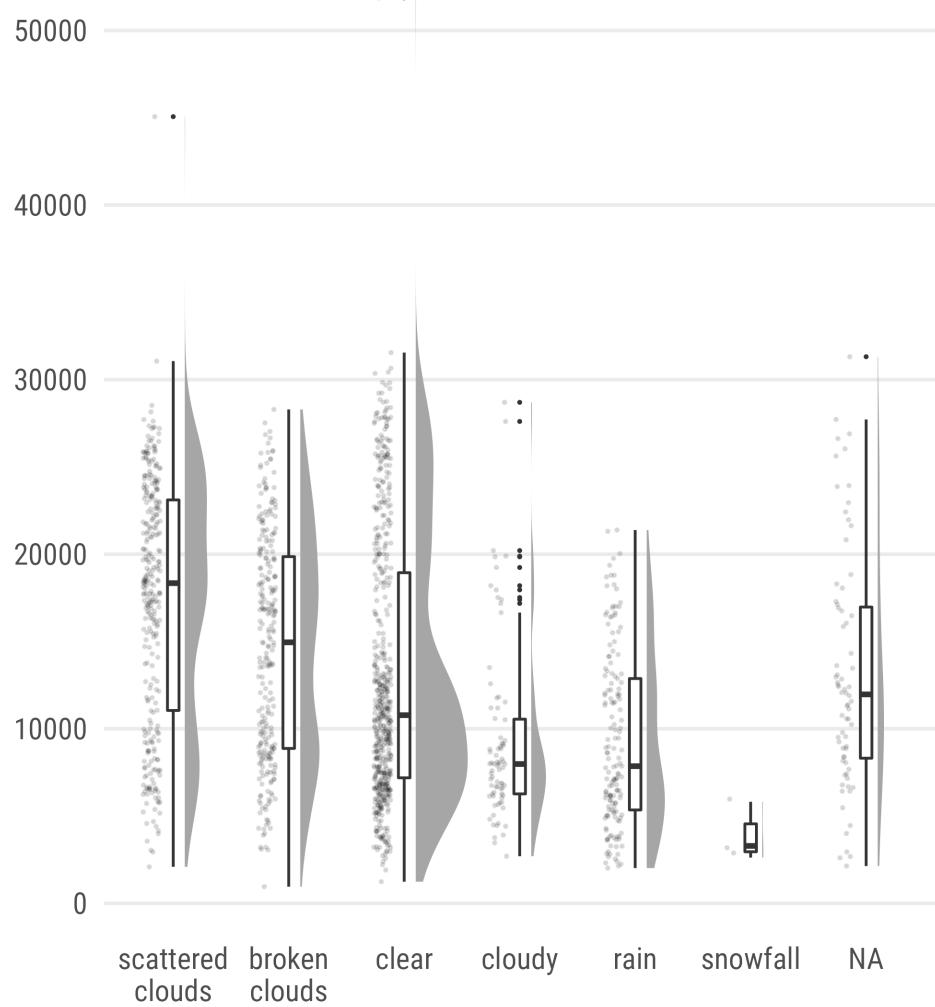
Reported bike shares by weather type



Raincloud Plots with {gghalves}

```
1 g +
2   ggdist::stat_halfeye(
3     aes(thickness = stat(f*n)),
4     color = NA,
5     width = .5,
6     position = position_nudge(x = .1)
7   ) +
8   geom_boxplot(
9     width = .1,
10    outlier.size = .2
11  ) +
12  gghalves::geom_half_point(
13    side = "l",
14    range_scale = .4,
15    size = .3,
16    alpha = .1
17  )
```

Reported bike shares by weather type



Raincloud Plots with {gghalves}

```
1 g +  
2   ggdist::stat_halfeye(  
3     aes(thickness = stat(f*n)),  
4     color = NA,  
5     width = .5,  
6     position = position_nudge(x = .1)  
7   ) +  
8   geom_boxplot(  
9     width = .1,  
10    outlier.size = .2  
11  ) +  
12  gghalves::geom_half_point(  
13    side = "l",  
14    range_scale = .4,  
15    size = .3,  
16    alpha = .1  
17  ) +  
18  coord_flip()
```

Reported bike shares by weather type

