

Meme Dog Intelligence

Kesava

Report Last Run: 2021-10-10 18:10:11

Contents

1 Preface	3
2 Get data	3
3 Check	3
4 Get meme dogs data only	4
5 Visualize	4

1 Preface

Came across the ggdogs package and I wanted to plot something using the package. Here is my attempt using the data.world dataset of dog intelligence.

2 Get data

```
sql_stmt <- qry_sql("SELECT * FROM dog_intelligence")
query_results_df <- data.world::query(
  sql_stmt, "len/dog-size-intelligence-linked")

## Rows: 136 Columns: 5

## -- Column specification -----
## Delimiter: ","
## chr (2): breed, classification
## dbl (3): obey, reps_lower, reps_upper

##
## 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.
```

3 Check

```
query_results_df %>% head()

## # A tibble: 6 x 5
##   breed      classification  obey  reps_lower  reps_upper
##   <chr>      <chr>        <dbl>     <dbl>       <dbl>
## 1 Border Collie Brightest Dogs  0.95      1          4
## 2 Poodle      Brightest Dogs  0.95      1          4
## 3 German Shepherd Brightest Dogs  0.95      1          4
## 4 Golden Retriever Brightest Dogs  0.95      1          4
## 5 Doberman Pinscher Brightest Dogs  0.95      1          4
## 6 Shetland Sheepdog Brightest Dogs  0.95      1          4
```

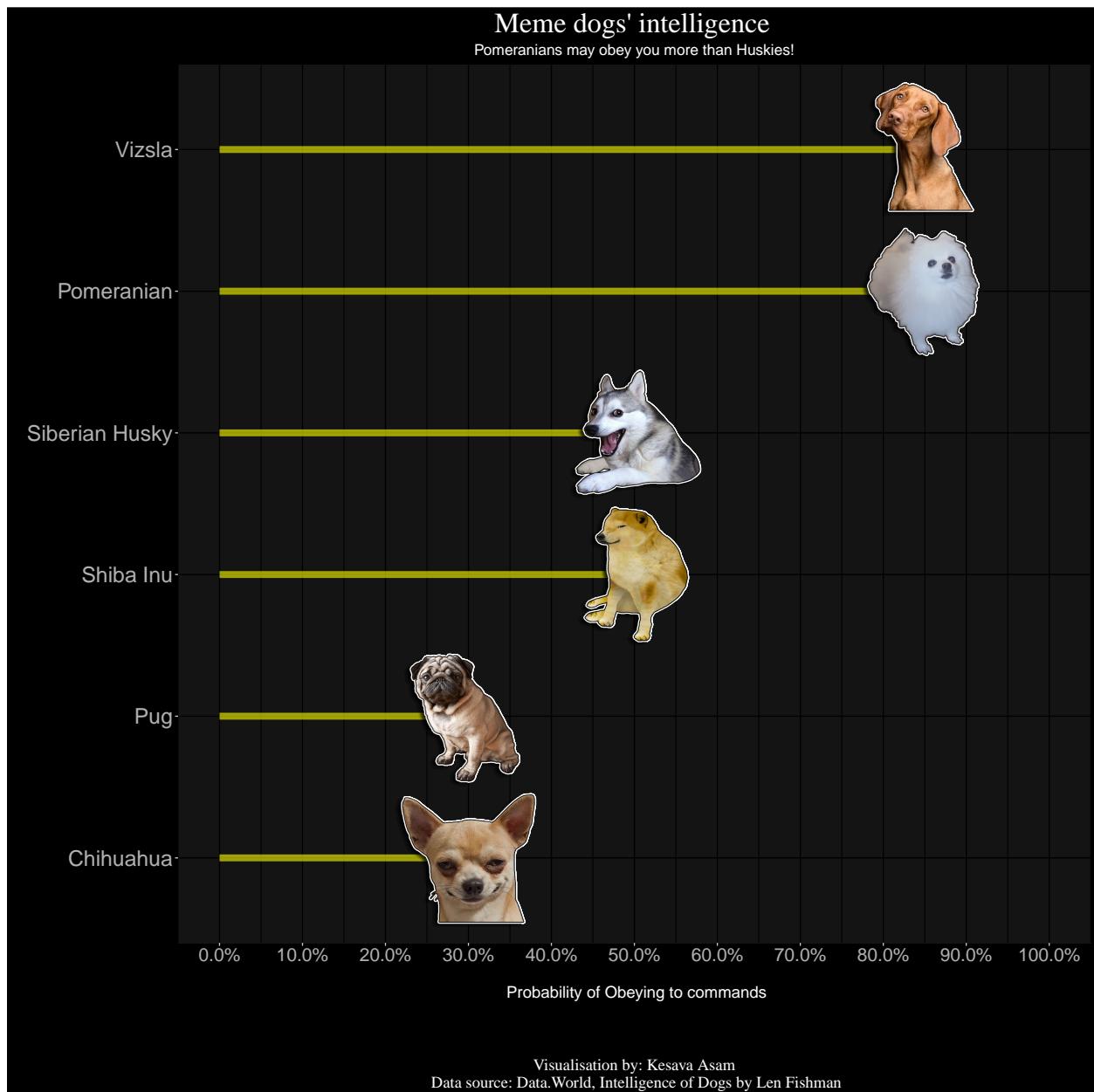
4 Get meme dogs data only

```
breed_int <- c("Siberian Husky", "Chihuahua", "Pug", "Shiba Inu", "Pomeranian",
             "Vizsla")
```

5 Visualize

```
query_results_df %>%
  filter(breed %in% breed_int) %>%
  mutate(breed = fct_reorder(breed, desc(reps_lower))) %>%
  # ggplot(aes(obey, breed)) +
  ggplot(aes(x = obey, xend = 0, y = breed, yend = breed)) +
  geom_segment(size = 3, color = "yellow", alpha = 0.6) +
  # geom_point(aes(x = reps_lower)) +
  geom_point() +
  geom_dog(aes(x = 0.30, y = 1), dog = "chihuahua", size = 4) +
  geom_dog(aes(x = 0.30, y = 2), dog = "pug", size = 4) +
  geom_dog(aes(x = 0.50, y = 3), dog = "doge", size = 4) +
  geom_dog(aes(x = 0.50, y = 4), dog = "husky", size = 4) +
  geom_dog(aes(x = 0.85, y = 5), dog = "gabe", size = 4) +
  geom_dog(aes(x = 0.85, y = 6), dog = "hearing", size = 4) +
  scale_x_continuous(labels = percent_format(),
                      breaks = seq(0,10, 0.1),
                      limits = c(0, 1)) +
  labs(x = "\n Probability of Obeying to commands",
       y = "",
       title = "Meme dogs' intelligence",
       subtitle = "Pomeranians may obey you more than Huskies!",
       #title = "Siberian Husky is not as intelligent as a pomeranian!!"
       caption = "\n\n\n Visualisation by: Kesava Asam \n Data source: Data.World, Intelligence of Dogs
theme_minimal() +
ggdark::dark_mode() +
theme(plot.title = element_text(hjust = 0.5, family = "serif", size = 26),
      plot.subtitle = element_text(hjust = 0.5, size = 14),
      plot.caption = element_text(hjust = 0.5, size = 15, family = "serif"),
      axis.title.x = element_text(size = 15),
      axis.text.x = element_text(size = 17),
      axis.text.y = element_text(size = 20))

## Inverted geom defaults of fill and color/colour.
## To change them back, use invert_geom_defaults().
```



```

sessionInfo()

## R version 4.1.1 (2021-08-10)
## Platform: x86_64-apple-darwin17.0 (64-bit)
## Running under: macOS Big Sur 10.16
##
## Matrix products: default
## BLAS:    /Library/Frameworks/R.framework/Versions/4.1/Resources/lib/libRblas.0.dylib
## LAPACK:  /Library/Frameworks/R.framework/Versions/4.1/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats      graphics   grDevices utils      datasets   methods    base
##
## other attached packages:
## [1] ggdogs_1.0       scales_1.1.1     data.world_1.3.1 dwapi_0.3.2
## [5] forcats_0.5.1    stringr_1.4.0    dplyr_1.0.7      purrr_0.3.4
## [9] readr_2.0.2      tidyr_1.1.4     tibble_3.1.5     ggplot2_3.3.5
## [13] tidyverse_1.3.1
##
## loaded via a namespace (and not attached):
## [1] Rcpp_1.0.7        lubridate_1.7.10 assertthat_0.2.1 digest_0.6.28
## [5] utf8_1.2.2       R6_2.5.1        cellranger_1.1.0 ggdark_0.2.1
## [9] backports_1.2.1  reprex_2.0.1    evaluate_0.14   highr_0.9
## [13] httr_1.4.2       pillar_1.6.3    rlang_0.4.11   curl_4.3.2
## [17] readxl_1.3.1    rstudioapi_0.13 magick_2.7.3   rmarkdown_2.11
## [21] bit_4.0.4        munsell_0.5.0   broom_0.7.9    compiler_4.1.1
## [25] modelr_0.1.8    xfun_0.26      pkgconfig_2.0.3 htmltools_0.5.2
## [29] tidyselect_1.1.1 fansi_0.5.0    crayon_1.4.1   tzdb_0.1.2
## [33] dbplyr_2.1.1    withr_2.4.2    grid_4.1.1    jsonlite_1.7.2
## [37] gtable_0.3.0    lifecycle_1.0.1 DBI_1.1.1    magrittr_2.0.1
## [41] cli_3.0.1       stringi_1.7.5  vroom_1.5.5   farver_2.1.0
## [45] fs_1.5.0        ini_0.3.1     xml2_1.3.2   ellipsis_0.3.2
## [49] generics_0.1.0   vctrs_0.3.8    tools_4.1.1   bit64_4.0.5
## [53] glue_1.4.2      hms_1.1.1     parallel_4.1.1 fastmap_1.1.0
## [57] yaml_2.2.1      colorspace_2.0-2 rvest_1.0.1  knitr_1.36
## [61] haven_2.4.3

```