Report

2025-11-13

```r
df = read.csv("./Datasets/MiWay_Bus_Stops_-4490514804018385747.csv")
head(df)
```

```
##   OBJECTID Stop_Number                           Stop_Description Shelter
## 1        1           1 ERIN MILLS PKY north of LINCOLN GREEN WAY        0
## 2        2           2         ERIN MILLS PKY at PAULA CRT WALKWAY        0
## 3        3           4           EGLINTON AVE west of EAST MILL RD        0
## 4        4           9         THE COLLEGEWAY east of GLEN ERIN DR        1
## 5        5          10            EXPLORER DR at 5875 EXPLORER DR        0
## 6        6          11        MCLAUGHLIN RD at SOUTH CEREMONIAL DR        1
##   Boarding_Allowed Debarking_Allowed Express_Stop Accessible Routes_Service
## 1                1                 1            0          1             13
## 2                1                 1            0          1             13
## 3                1                 1            0          1             35
## 4                1                 1            0          1          29/36
## 5                1                 1            0          0             74
## 6                1                 1            0          1             66
##             District            Zone Latitude Longitude Effective_Date
## 1           SHERIDAN  TRAFFIC ZONE 51 43.53525 -79.66364     2025/09/01
## 2           SHERIDAN  TRAFFIC ZONE 50 43.53453 -79.66312     2025/09/01
## 3        EAST CREDIT  TRAFFIC ZONE 88 43.57937 -79.68393     2025/09/01
## 4          ERIN MILLS TRAFFIC ZONE 213 43.53722 -79.68434     2025/09/01
## 5 AIRPORT CORPORATE   TRAFFIC ZONE 5 43.65462 -79.61673     2025/09/01
## 6         HURONTARIO  TRAFFIC ZONE 89 43.59985 -79.66698     2025/09/01
##           x        y
## 1 -79.66364 43.53525
## 2 -79.66312 43.53453
## 3 -79.68393 43.57937
## 4 -79.68434 43.53722
## 5 -79.61673 43.65462
## 6 -79.66698 43.59985
```

```r
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```
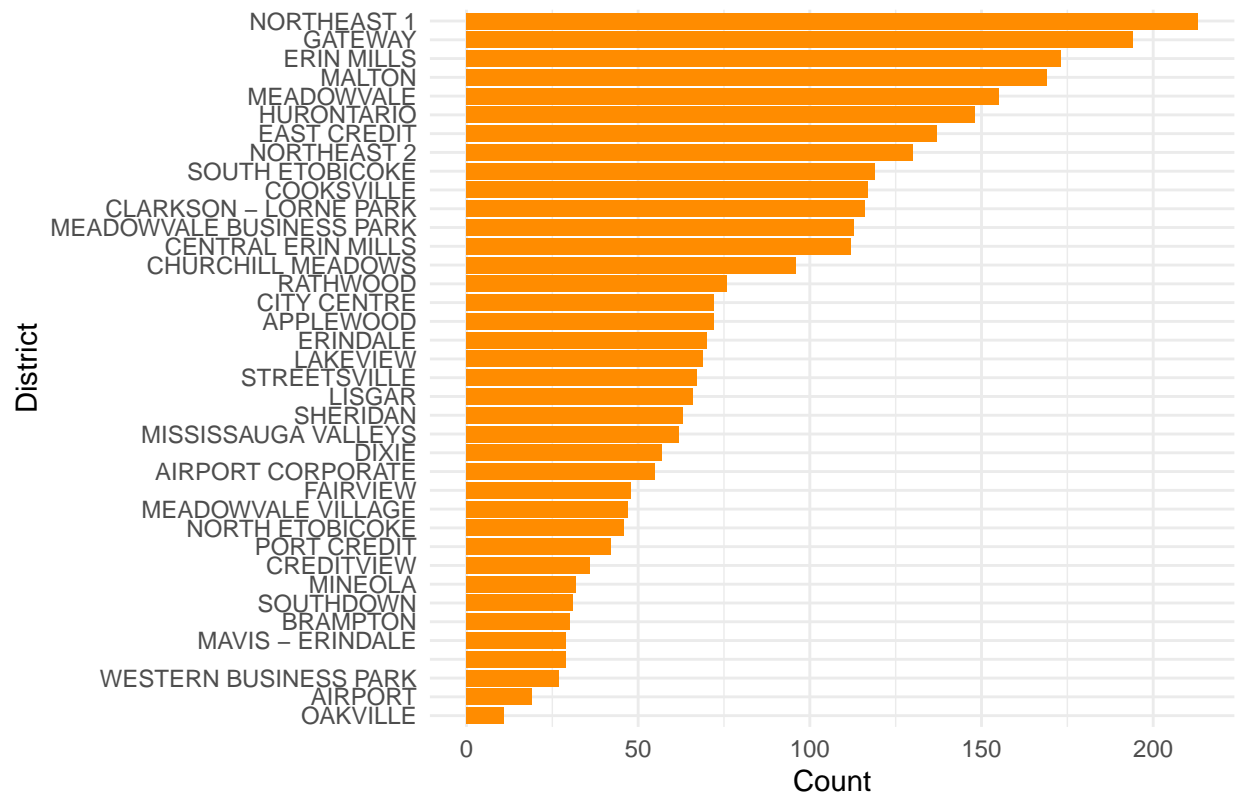
```r
library(ggplot2)
# install.packages("dplyr")
# install.packages("ggplot2")
counts <- df %>%
  group_by(District) %>%
  summarise(n = n()) %>%
  arrange(desc(n))
head(counts)
```

```
## # A tibble: 6 x 2
##   District       n
##   <chr>      <int>
## 1 NORTHEAST 1  213
## 2 GATEWAY      194
## 3 ERIN MILLS   173
## 4 MALTON       169
## 5 MEADOWVALE   155
## 6 HURONTARIO   148
```

```r
df <- df %>% filter(District != "", District != "UNKNOWN")
```

```r
ggplot(counts, aes(x = n, y = reorder(District, n))) +
  geom_col(fill = "darkorange") +
  labs(
    title = "Number of Stops by District",
    x = "Count",
    y = "District"
  ) +
  theme_minimal()
```

Number of Stops by District

```r
unique(df$District)
```

```
##  [1] "SHERIDAN"                 "EAST CREDIT"
##  [3] "ERIN MILLS"               "AIRPORT CORPORATE"
##  [5] "HURONTARIO"               "NORTHEAST 1"
##  [7] "MEADOWVALE"               "GATEWAY"
##  [9] "LISGAR"                   "STREETSVILLE"
## [11] "CENTRAL ERIN MILLS"       "MISSISSAUGA VALLEYS"
## [13] "MEADOWVALE BUSINESS PARK" "FAIRVIEW"
## [15] "CITY CENTRE"              "COOKSVILLE"
## [17] "MEADOWVALE VILLAGE"       "CLARKSON – LORNE PARK"
## [19] "OAKVILLE"                 "SOUTHDOWN"
## [21] "PORT CREDIT"              "RATHWOOD"
## [23] "MINEOLA"                  "NORTHEAST 2"
## [25] "SOUTH ETOBICOKE"          "AIRPORT"
## [27] "LAKEVIEW"                 "NORTH ETOBICOKE"
## [29] "CHURCHILL MEADOWS"        "DIXIE"
## [31] "ERINDALE"                 "MAVIS – ERINDALE"
## [33] "CREDITVIEW"               "APPLEWOOD"
## [35] "MALTON"                   "WESTERN BUSINESS PARK"
## [37] "BRAMPTON"
```

```r
access_summary <- df %>%
  group_by(District) %>%
  summarise(
```
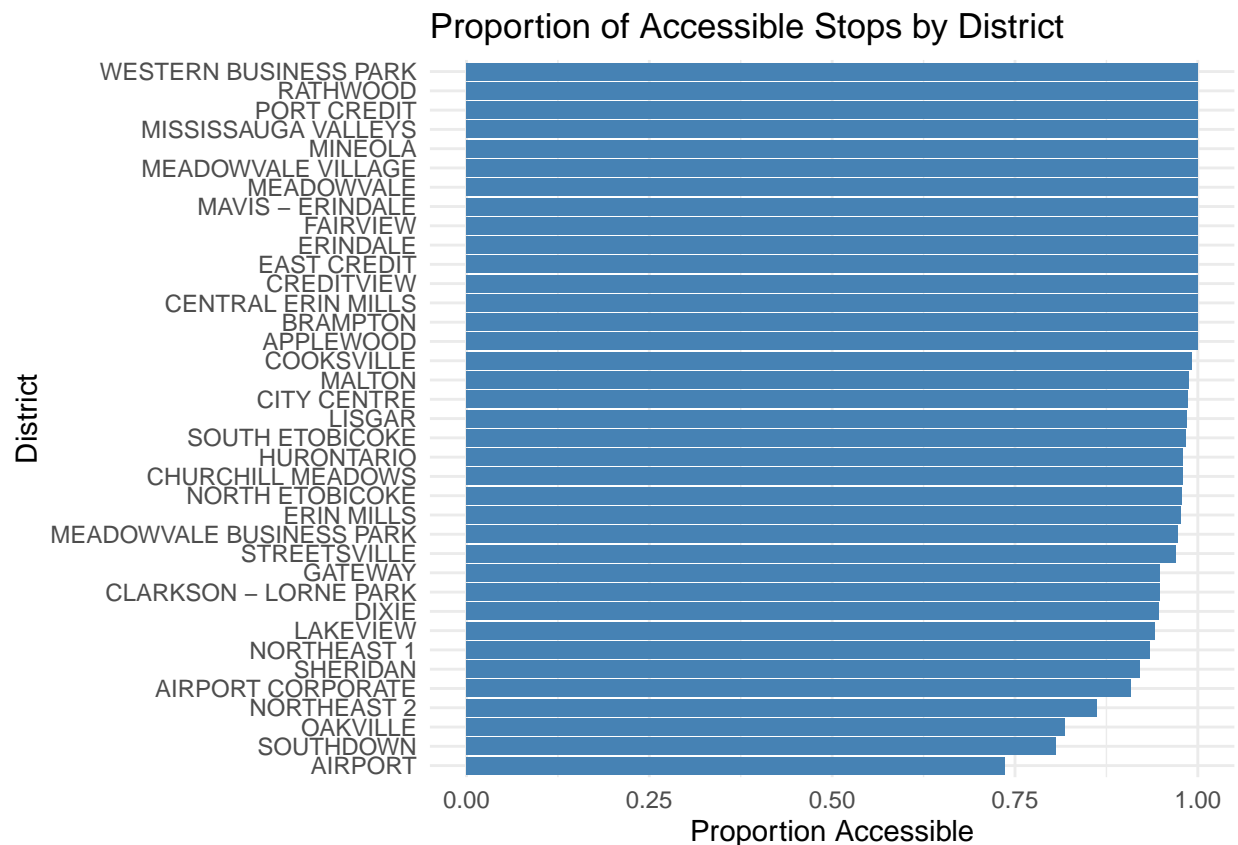
```
    accessible = sum(Accessible),
    total = n(),
    proportion = accessible / total
  ) %>%
  arrange(desc(proportion))
```

```
ggplot(access_summary, aes(x = proportion, y = reorder(District, proportion))) +
  geom_col(fill = "steelblue") +
  labs(
    title = "Proportion of Accessible Stops by District",
    x = "Proportion Accessible",
    y = "District"
  ) +
  theme_minimal()
```
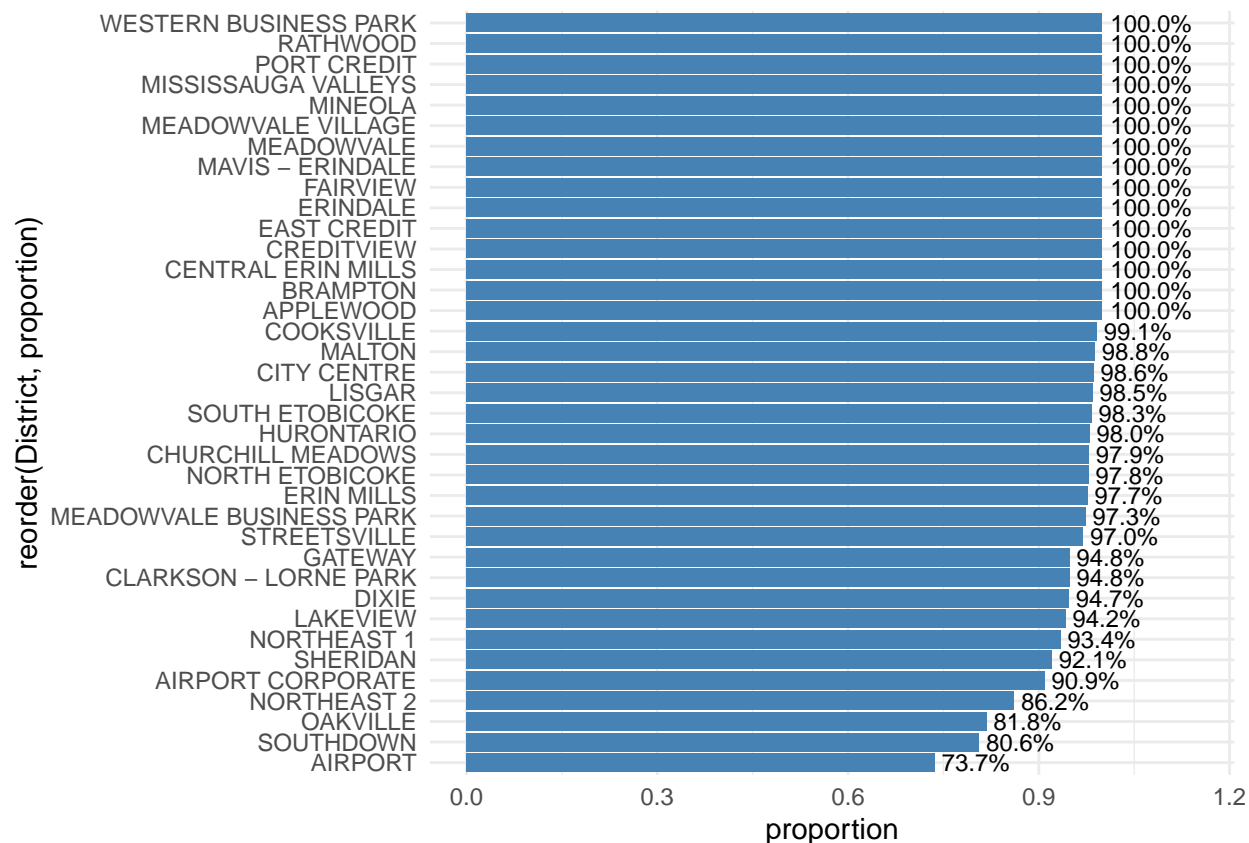

Proportion of Accessible Stops by District

AODA (Accessibility for Ontarians with Disabilities Act) Requirements

```
ggplot(access_summary, aes(x = proportion, y = reorder(District, proportion))) +
  geom_col(fill = "steelblue") +
  geom_text(aes(label = sprintf("%.1f%%", proportion * 100)),
            hjust = -0.1, size = 3) +
  xlim(0, 1.15) +
  theme_minimal()
```
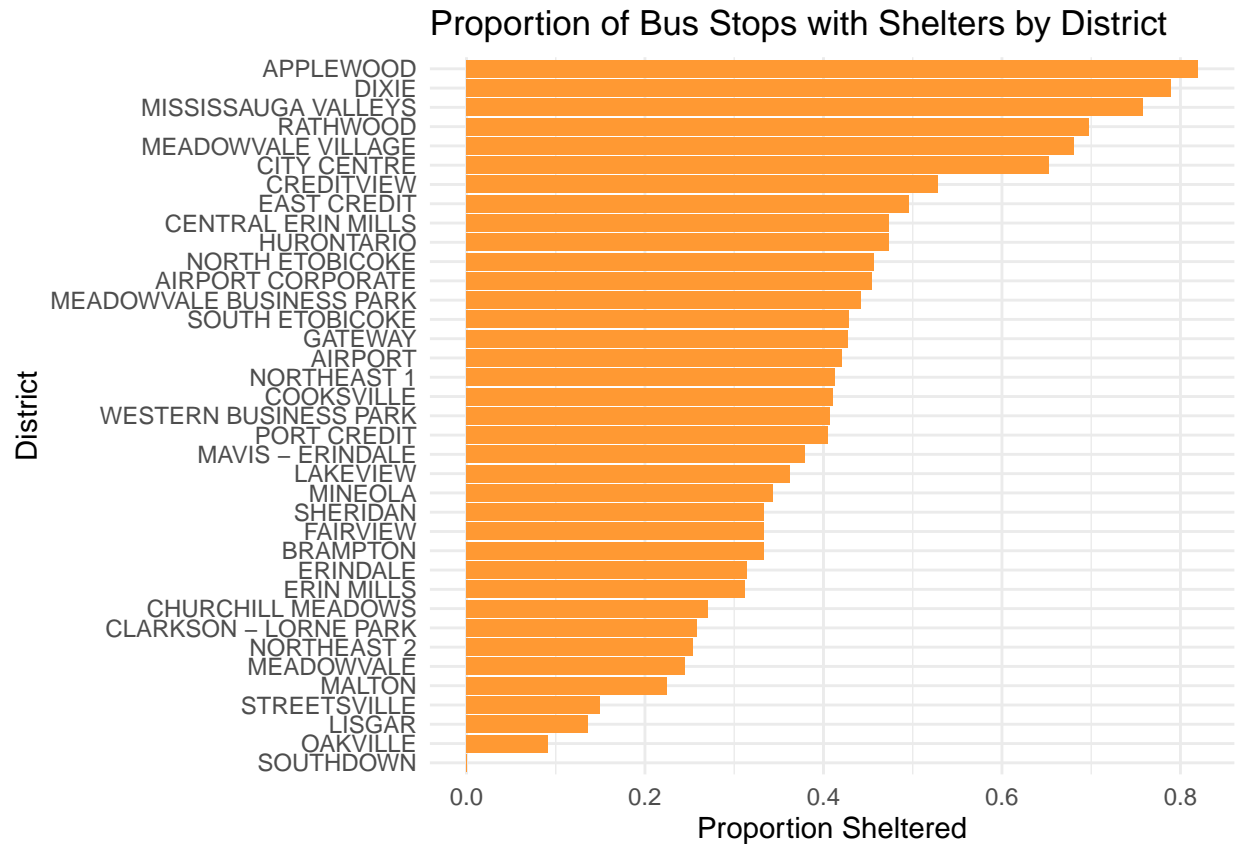
Shelter

```r
shelter_summary <- df %>%
  group_by(District) %>%
  summarise(
    shelters = sum(Shelter),
    total = n(),
    proportion_sheltered = shelters / total
  ) %>%
  arrange(desc(proportion_sheltered))
head(shelter_summary)
```

```
## # A tibble: 6 x 4
##   District             shelters total proportion_sheltered
##   <chr>                   <int> <int>                <dbl>
## 1 APPLEWOOD                  59    72                0.819
## 2 DIXIE                      45    57                0.789
## 3 MISSISSAUGA VALLEYS        47    62                0.758
## 4 RATHWOOD                   53    76                0.697
## 5 MEADOWVALE VILLAGE         32    47                0.681
## 6 CITY CENTRE                47    72                0.653
```
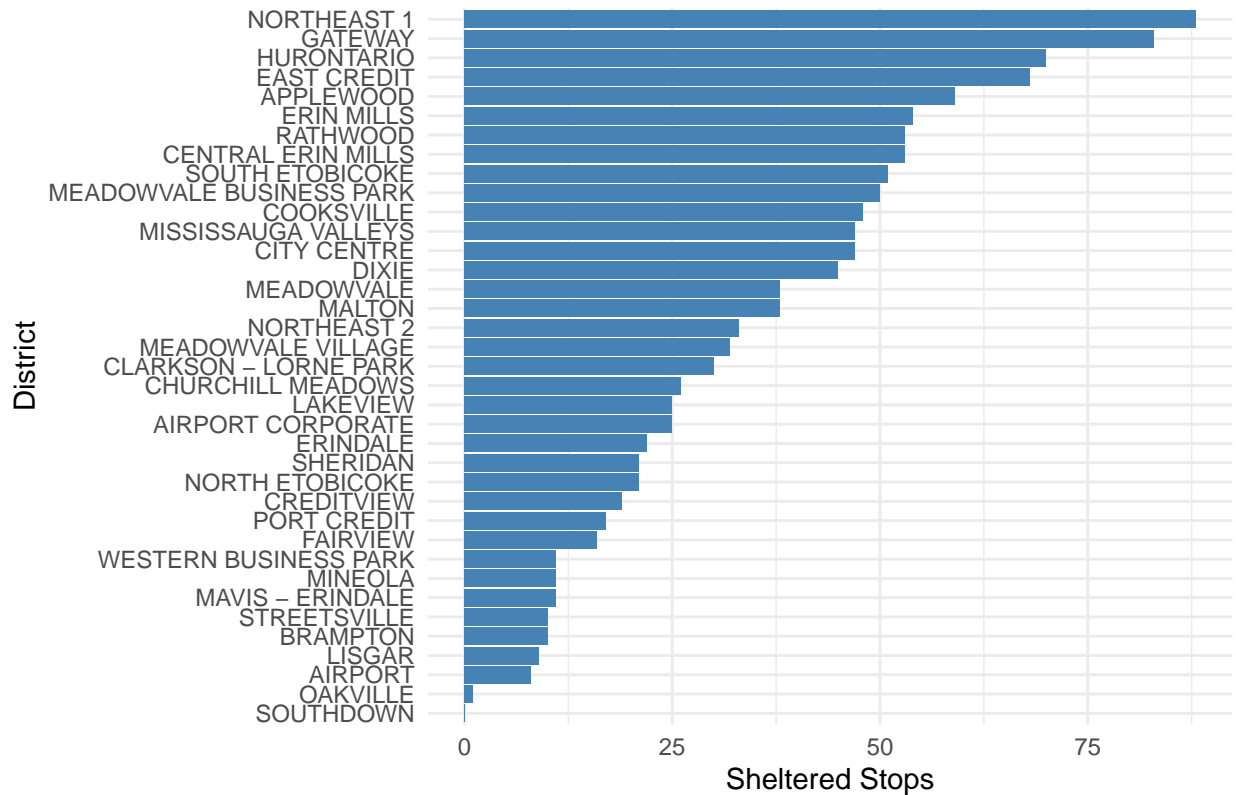
```r
ggplot(shelter_summary, aes(x = proportion_sheltered,
                            y = reorder(District, proportion_sheltered))) +
  geom_col(fill = "#FF9933") +
```

```
labs(
  title = "Proportion of Bus Stops with Shelters by District",
  x = "Proportion Sheltered",
  y = "District"
) +
theme_minimal()
```

Proportion of Bus Stops with Shelters by District


```
ggplot(shelter_summary, aes(x = shelters, y = reorder(District, shelters))) +
  geom_col(fill = "steelblue") +
  labs(
    title = "Number of Sheltered Stops by District",
    x = "Sheltered Stops",
    y = "District"
  ) +
  theme_minimal()
```

Number of Sheltered Stops by District


```r
library(dplyr)
library(tidyr)

routes_long <- df %>%
  separate_rows(Routes_Service, sep = "/") %>%
  mutate(Routes_Service = trimws(Routes_Service)) %>%
  filter(Routes_Service != "")
unique_buses <- unique(routes_long$Routes_Service)
unique_buses
```

```
##  [1] "13"  "35"  "29"  "36"  "74"  "66"  "7"   "39"  "43"  "10"  "313" "107"
## [13] "109" "73"  "25"  "306" "9"   "48"  "3"   "53"  "108" "70"  "8"   "28"
## [25] "26"  "110" "61"  "90"  "38"  "17"  "103" "45"  "14"  "1"   "23"  "51"
## [37] "302" "2"   "15"  "5"   "46"  "49"  "18"  "42"  "6"   "11"  "30"  "4"
## [49] "31"  "312" "68"  "314" "20"  "315" "71"  "44"  "126" "101" "24"  "57"
## [61] "307" "16"  "304" "22"  "321"
```
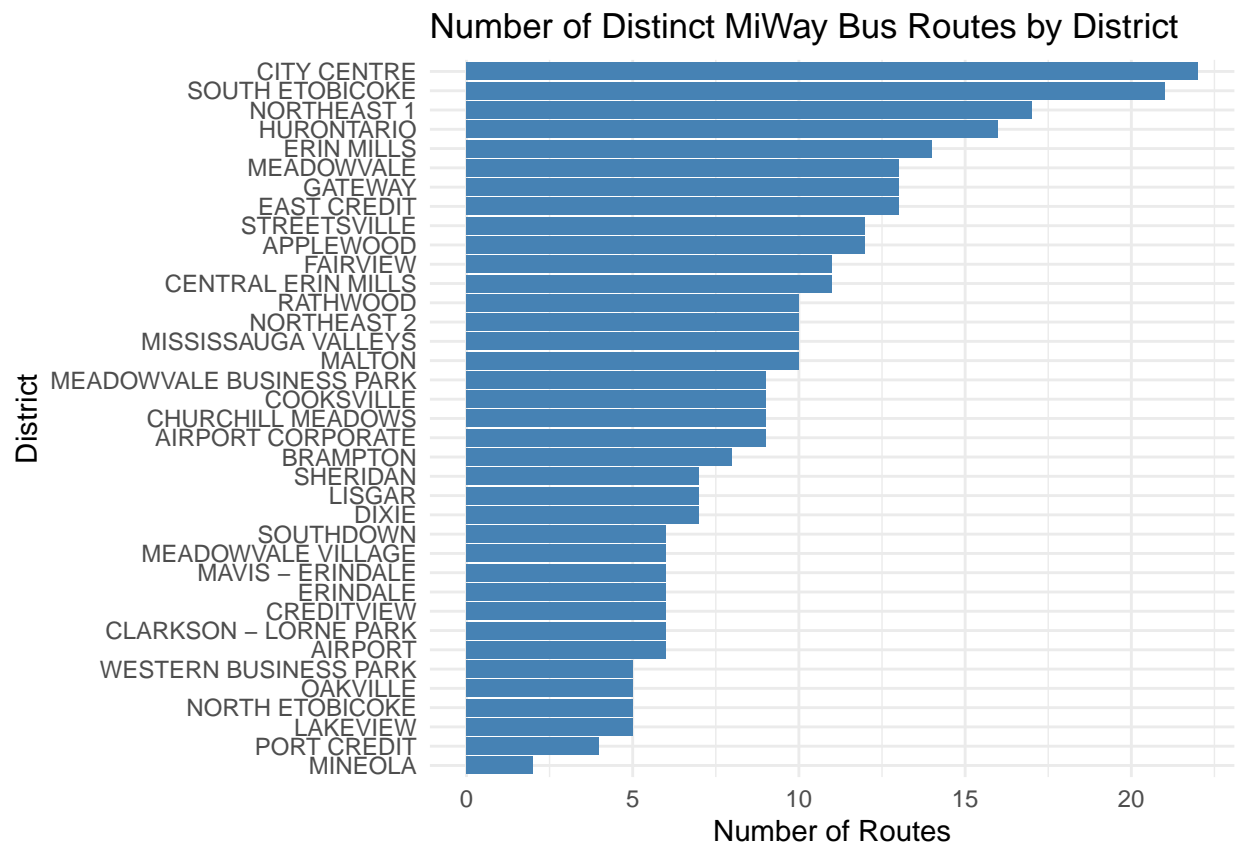
```r
routes_by_district <- routes_long %>%
  group_by(District) %>%
  summarise(n_routes = n_distinct(Routes_Service)) %>%
  arrange(desc(n_routes))
routes_by_district
```

```
## # A tibble: 37 x 2
##    District        n_routes
```

```
##    <chr>              <int>
##  1 CITY CENTRE           22
##  2 SOUTH ETOBICOKE       21
##  3 NORTHEAST 1           17
##  4 HURONTARIO            16
##  5 ERIN MILLS            14
##  6 EAST CREDIT           13
##  7 GATEWAY               13
##  8 MEADOWVALE            13
##  9 APPLEWOOD             12
## 10 STREETSVILLE          12
## # i 27 more rows
```

```r
library(ggplot2)

ggplot(routes_by_district, aes(x = n_routes, y = reorder(District, n_routes))) +
  geom_col(fill = "steelblue") +
  labs(
    title = "Number of Distinct MiWay Bus Routes by District",
    x = "Number of Routes",
    y = "District"
  ) +
  theme_minimal()
```


Number of Distinct MiWay Bus Routes by District

```r
utm_routes <- c("1", "44", "101", "110", "126")
unique(df$Routes_Service)
```

```
##   [1] "13"
##   [2] "35"
##   [3] "29/36"
##   [4] "74"
##   [5] "66"
##   [6] "7/74"
##   [7] "7/39/43"
##   [8] "10/313"
##   [9] "43"
##  [10] "39/43"
##  [11] "107/109"
##  [12] "39/43/73"
##  [13] "25/43"
##  [14] "39"
##  [15] "10/39/43/306"
##  [16] "9/48"
##  [17] "9/13/48"
##  [18] "3/53"
##  [19] "108"
##  [20] "25/70"
##  [21] "3/8"
##  [22] "28"
##  [23] "3"
##  [24] "26/110"
##  [25] "61"
##  [26] "39/90"
##  [27] "38/39/43/90"
##  [28] "17"
##  [29] "17/103"
##  [30] "13/29/45"
##  [31] "13/14/45/110"
##  [32] "13/14/45"
##  [33] "13/14/29/45/110"
##  [34] "29"
##  [35] "29/45"
##  [36] "1/38"
##  [37] "53"
##  [38] "39/53"
##  [39] "23"
##  [40] "14/29"
##  [41] "14/23"
##  [42] "14"
##  [43] "51/302"
##  [44] "7"
##  [45] "48"
##  [46] "2"
##  [47] "15"
##  [48] "5"
##  [49] "26"
##  [50] "2/8/14/23"
```

```
##  [51] "13/46/49"
##  [52] "45"
##  [53] "7/35"
##  [54] "2/8"
##  [55] "18/42"
##  [56] "8"
##  [57] "10/45/313"
##  [58] "13/29"
##  [59] "6"
##  [60] "11/30"
##  [61] "46"
##  [62] "4"
##  [63] "23/31"
##  [64] "31"
##  [65] "4/5/31"
##  [66] "4/5"
##  [67] "5/23/31"
##  [68] "5/23"
##  [69] "4/31"
##  [70] "61/312"
##  [71] "68/314"
##  [72] "2/3/7/10/17/20/26/68/315"
##  [73] "2/3/10/26"
##  [74] "45/71"
##  [75] "44/126"
##  [76] "68/315"
##  [77] "1/29/101"
##  [78] "13/71"
##  [79] "1"
##  [80] "13/110"
##  [81] "36"
##  [82] "1/101/110"
##  [83] "1/101"
##  [84] "29/71"
##  [85] "13/29/110"
##  [86] "73"
##  [87] "24/107"
##  [88] "6/8/26/312"
##  [89] "6/8/26/126/312"
##  [90] "8/26/126/302/312"
##  [91] "8/26/312"
##  [92] "26/302/312"
##  [93] "10"
##  [94] "8/9/10"
##  [95] "35/48"
##  [96] "1/38/101"
##  [97] "70"
##  [98] "6/38"
##  [99] "38"
## [100] "9/35"
## [101] "2/103"
## [102] "2/4"
## [103] "2/4/103"
## [104] "4/28"
```

```
## [105] "7/20/315"
## [106] "2/53/103"
## [107] "45/109"
## [108] "45/49/109"
## [109] "24/43/57/74"
## [110] "13/48"
## [111] "44"
## [112] "10/39/43"
## [113] "315"
## [114] "51"
## [115] "1/3/26/307"
## [116] "3/26"
## [117] "314/315"
## [118] "1/11/20/35/70/101/108/109/126"
## [119] "5/31"
## [120] "9"
## [121] "9/61"
## [122] "42/90"
## [123] "49"
## [124] "38/306/314"
## [125] "16"
## [126] "43/46"
## [127] "24"
## [128] "1/36"
## [129] "26/126"
## [130] "20"
## [131] "13/26/29"
## [132] "1/13/36/48"
## [133] "13/101"
## [134] "1/45"
## [135] "1/36/48"
## [136] "110"
## [137] "7/39/107/109"
## [138] "2/53"
## [139] "6/26"
## [140] "57/61"
## [141] "68/306/314"
## [142] "68/306/315"
## [143] "6/26/126"
## [144] "1/6"
## [145] "2/302"
## [146] "2/103/302"
## [147] "304"
## [148] "8/53"
## [149] "8/53/304"
## [150] "8/304"
## [151] "39/43/314"
## [152] "3/302/304"
## [153] "302/304"
## [154] "26/126/312"
## [155] "26/312"
## [156] "90"
## [157] "1/3/307"
## [158] "3/307"
```

```
## [159] "51/307"
## [160] "20/26"
## [161] "51/302/307"
## [162] "3/302"
## [163] "42"
## [164] "20/26/126"
## [165] "2/17"
## [166] "68/107/109"
## [167] "107"
## [168] "26/109"
## [169] "9/38"
## [170] "6/8/26/302/312"
## [171] "38/108"
## [172] "6/8/9/26/28/61/66/302/312"
## [173] "10/38/39/43/45/46/90/313"
## [174] "38/43"
## [175] "10/53"
## [176] "28/38"
## [177] "71"
## [178] "9/46/48"
## [179] "39/46"
## [180] "48/110"
## [181] "68"
## [182] "48/313"
## [183] "10/45/109/313"
## [184] "46/48"
## [185] "25/43/70"
## [186] "25"
## [187] "2/10"
## [188] "17/70"
## [189] "9/28/61/66"
## [190] "10/306"
## [191] "10/39"
## [192] "103"
## [193] "17/70/103"
## [194] "35/68"
## [195] "5/73"
## [196] "5/73/74"
## [197] "7/35/74"
## [198] "7/43/74"
## [199] "57"
## [200] "38/313"
## [201] "38/314"
## [202] "17/53"
## [203] "44/90"
## [204] "44/48/90"
## [205] "10/109"
## [206] "3/8/302"
## [207] "38/44/108"
## [208] "38/43/44/108"
## [209] "7/30"
## [210] "38/68"
## [211] "14/45"
## [212] "38/43/108"
```

```
## [213] "44/306"
## [214] "43/44"
## [215] "49/313"
## [216] "313"
## [217] "5/15"
## [218] "42/108"
## [219] "15/18/42"
## [220] "15/51"
## [221] "25/39/53"
## [222] "38/39/43/46/90"
## [223] "4/28/103"
## [224] "39/57"
## [225] "23/29"
## [226] "15/30"
## [227] "22"
## [228] "24/30"
## [229] "11/18/42"
## [230] "18/30/42"
## [231] "30"
## [232] "7/15/30"
## [233] "11/18/24/42"
## [234] "11/18/42/107"
## [235] "11"
## [236] "15/24"
## [237] "16/22"
## [238] "39/43/68/314"
## [239] "7/24/107"
## [240] "7/24"
## [241] "39/43/306"
## [242] "39/70"
## [243] "66/70"
## [244] "4/28/38"
## [245] "9/61/66"
## [246] "49/321"
## [247] "73/74/107/109"
## [248] "306/313"
## [249] "7/39/43/74"
## [250] "18/42/107"
## [251] "43/68/314"
## [252] "321"
## [253] "46/49"
## [254] "9/13/46/49"
## [255] "109/110"
## [256] "45/49"
## [257] "35/321"
## [258] "9/321"
## [259] "101"
## [260] "36/45"
## [261] "109"
## [262] "9/49"
## [263] "49/306"
## [264] "46/321"
## [265] "3/6/7/8/9/20/28/66/68/107/302/312/315"
## [266] "10/13/38/39/42/43/44/45/46/48/90/109/313"
```

```
## [267] "18/66"
## [268] "61/66"
## [269] "18/57"
## [270] "18"
## [271] "1/11/20/35/70"
## [272] "1/3/26/71/307"
## [273] "26/71"
## [274] "70/71/108/307"
## [275] "126"
```
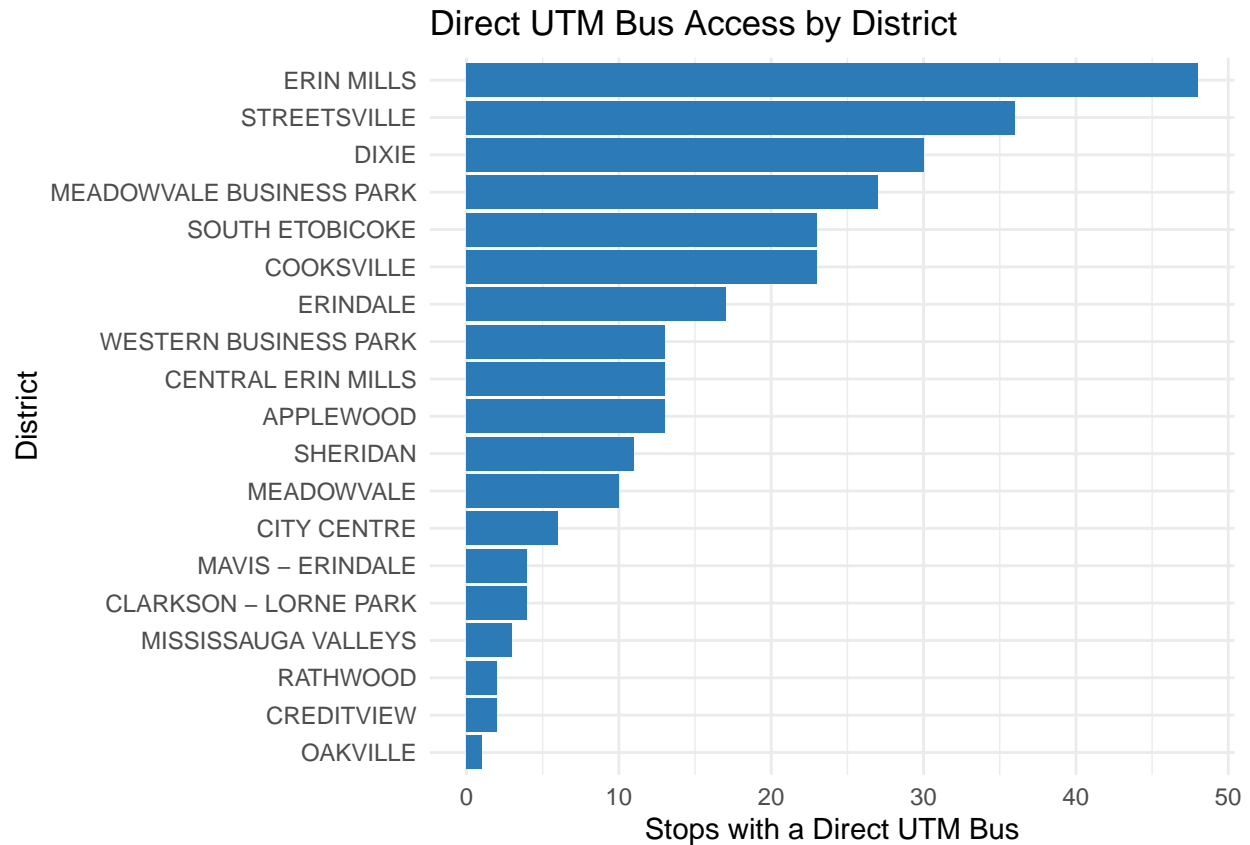
```r
library(dplyr)
library(tidyr)

routes_long <- df %>%
  separate_rows(Routes_Service, sep = "/") %>%
  mutate(Routes_Service = trimws(Routes_Service))
```

```r
utm_long <- routes_long %>%
  filter(Routes_Service %in% utm_routes)

district_utm <- utm_long %>%
  group_by(District) %>%
  summarise(
    utm_stops          = n_distinct(Stop_Number),    # stops with a direct UTM bus
    utm_bus_occurrences = n(),                        # total appearances of UTM routes
    utm_distinct_routes = n_distinct(Routes_Service)  # how many of the 5 show up here
  ) %>%
  arrange(desc(utm_stops))
```

```r
ggplot(district_utm, aes(x = utm_stops, y = reorder(District, utm_stops))) +
  geom_col(fill = "#2C7BB6") +
  labs(
    title = "Direct UTM Bus Access by District",
    x = "Stops with a Direct UTM Bus",
    y = "District"
  ) +
  theme_minimal()
```

Direct UTM Bus Access by District


```r
# One row per (Stop_Number, District, has_utm)
stop_level <- routes_long %>%
  group_by(Stop_Number, District) %>%
  summarise(
    has_utm = any(Routes_Service %in% utm_routes),
    .groups = "drop"
  )

utm_normalized <- stop_level %>%
  group_by(District) %>%
  summarise(
    total_stops = n(),
    utm_stops   = sum(has_utm),
    utm_rate    = utm_stops / total_stops    # THIS IS normalization
  ) %>%
  arrange(desc(utm_rate))
```
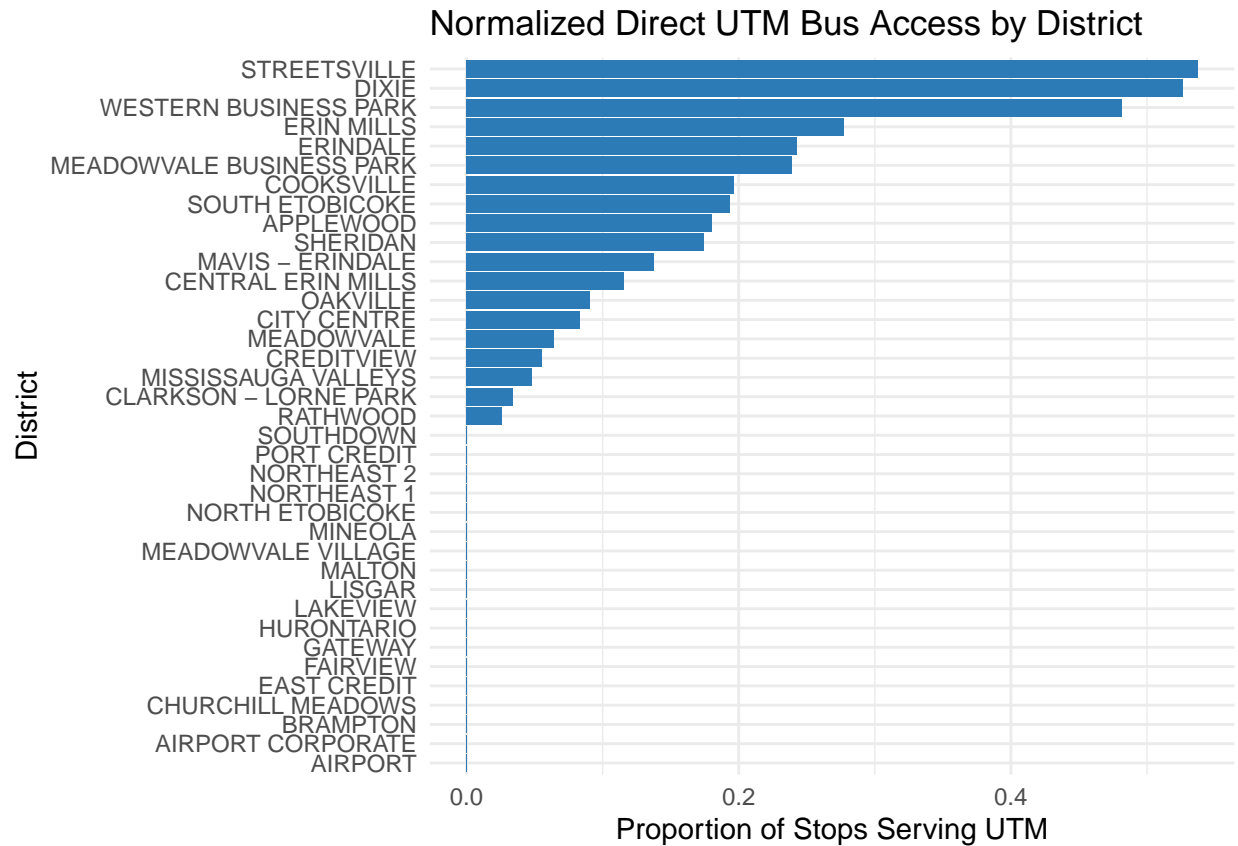
```r
ggplot(utm_normalized, aes(
    x = utm_rate,
    y = reorder(District, utm_rate)
)) +
  geom_col(fill = "#2C7BB6") +
  labs(
    title = "Normalized Direct UTM Bus Access by District",
    x = "Proportion of Stops Serving UTM",
```

```
    y = "District"
) +
theme_minimal()
```

Normalized Direct UTM Bus Access by District


```
go_df = read.csv("./Datasets/GO_Transit_Stations.csv")
stations = go_df$LANDMARKNAME
stations
```

```
## [1] "Clarkson Go Station"     "Cooksville Go Station"
## [3] "Dixie Go Station"        "Erindale Go Station"
## [5] "Lisgar Go Station"       "Malton Go Station"
## [7] "Meadowvale Go Station"   "Port Credit Go"
## [9] "Streetsville Go Station"
```

```
# Tag GO-station stops in the MiWay stops data
df$is_GO_stop <- grepl("GO STATION", df$Stop_Description, ignore.case = TRUE)

# Explode routes into one row per (stop, route)
routes_long <- df %>%
  separate_rows(Routes_Service, sep = "/") %>%
  mutate(Routes_Service = trimws(Routes_Service))

# All route numbers that serve at least one GO station stop
go_routes <- routes_long %>%
  filter(is_GO_stop, Routes_Service != "") %>%    # keep only GO stops + non-empty routes
```

16

```
  pull(Routes_Service) %>%
  unique()

# For each (real) district, how many distinct GO-serving routes & GO-connected stops?
go_by_district <- routes_long %>%
  filter(
    Routes_Service %in% go_routes,        # only routes that touch a GO station
    District != "",                       # drop blank districts
    District != "UNKNOWN"                 # drop UNKNOWN pseudo-district
  ) %>%
  group_by(District) %>%
  summarise(
    distinct_go_routes = n_distinct(Routes_Service),
    total_go_stops     = n_distinct(Stop_Number),
    .groups = "drop"
  ) %>%
  arrange(desc(distinct_go_routes))
```

```
ggplot(go_by_district, aes(
    x = distinct_go_routes,
    y = reorder(District, distinct_go_routes)
)) +
  geom_col(fill = "#2C7BB6") +
  labs(
    title = "MiWay Route Connectivity to GO Stations by District",
    x = "Distinct GO-Serving Bus Routes",
    y = "District"
  ) +
  theme_minimal()
```

MiWay Route Connectivity to GO Stations by District