Book Data Linkage Statistics

This notebook presents statistics of the book data integration.

Setup

```
library(tidyverse, warn.conflicts=FALSE)
library(arrow, warn.conflicts=FALSE)
library(jsonlite)
```

I want to use `theme_minimal()` by default:

```
theme_set(theme_minimal())
```

And default image sizes aren't great:

```
options(repr.plot.width  = 7,
        repr.plot.height = 4)
```

Load Link Stats

We compute dataset linking statistics as `gender-stats.csv` as part of the integration. Let's load those:

```
link_stats = read_csv("book-links/gender-stats.csv")
glimpse(link_stats)
```

```
Rows: 46 Columns: 4
── Column specification ────────────────────────────────────────────────────────
Delimiter: ","
chr (2): dataset, gender
dbl (2): n_books, n_actions

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.

Rows: 46
Columns: 4
$ dataset   <chr> "LOC-MDS", "LOC-MDS", "LOC-MDS", "LOC-MDS", "LOC-MDS", "LOC-…
```

```
$ gender    <chr> "male", "unknown", "no-author-rec", "ambiguous",
"no-book-au…
$ n_books   <dbl> 2424009, 1084460, 306292, 73989, 600214, 743105,
102756, 314…
$ n_actions <dbl> NA, NA, NA, NA, NA, NA, 468156, 69361, 401483,
47275, 104008…
```

Now let's define variables for our variou codes. We are first going to define our gender codes. We'll start with the resolved codes:

```
link_codes = c('female', 'male', 'ambiguous', 'unknown')
```

We want the unlink codes in order, so the last is the first link failure:

```
unlink_codes = c('no-author-rec', 'no-book-author', 'no-book')
```

```
all_codes = c(link_codes, unlink_codes)
```

Processing Statistics

Now we'll pivot each of our count columns into a table for easier reference.

```
book_counts = link_stats %>%
    pivot_wider(id_cols=dataset, names_from=gender, values_from=n_books) %>%
    replace(is.na(.), 0) %>%
    mutate(total=rowSums(across(-dataset)))
glimpse(book_counts)
```

```
Rows: 7
Columns: 9
$ dataset          <chr> "LOC-MDS", "BX-I", "BX-E", "AZ14",
"AZ18", "GR-I", "G…
$ male             <dbl> 2424009, 102756, 58484, 550877, 670899,
338411, 334136
$ unknown          <dbl> 1084460, 31440, 15281, 239915, 300300,
108333, 106501
$ `no-author-rec`  <dbl> 306292, 11562, 5692, 155511, 239917,
61601, 60515
$ ambiguous        <dbl> 73989, 9528, 5596, 24064, 27977, 18709,
18516
$ `no-book-author` <dbl> 600214, 10861, 5428, 167948, 152438,
750118, 738282
$ female           <dbl> 743105, 71441, 40256, 248863, 318004,
228142, 225840
$ `no-book`        <dbl> 0, 35009, 17481, 870268, 1144899, 0, 0
$ total            <dbl> 5232069, 272597, 148218, 2257446,
2854434, 1505314, 1…
```

```
act_counts = link_stats %>%
    filter(dataset != "LOC-MDS") %>%
    pivot_wider(id_cols=dataset, names_from=gender, values_from=n_actions) %>%
    replace(is.na(.), 0) %>%
```

```
    mutate(total=rowSums(across(-dataset)))
glimpse(act_counts)
```

```
Rows: 6
Columns: 9
$ dataset          <chr> "BX-I", "BX-E", "AZ14", "AZ18", "GR-I",
"GR-E"
$ male             <dbl> 468156, 183945, 7105363, 15603235,
69977512, 33249747
$ unknown          <dbl> 69361, 24554, 2157265, 4692726, 10242726,
3570086
$ female           <dbl> 401483, 142252, 4977284, 12377052,
82889862, 36335167
$ `no-book`        <dbl> 47275, 19920, 3879190, 10008921, 0, 0
$ ambiguous        <dbl> 104008, 41768, 849025, 1844630, 22091068,
13230835
$ `no-author-rec`  <dbl> 18597, 7130, 1100127, 3312340, 3545964,
1039410
$ `no-book-author` <dbl> 18882, 7234, 2359170, 2820794, 29784689,
11168052
$ total            <dbl> 1127762, 426803, 22427424, 50659698,
218531821, 98593…
```

We're going to want to compute versions of this table as fractions, e.g. the fraction of books that are written by women. We will use the following helper function:

```
fractionalize = function(data, columns, unlinked=NULL) {
    fracs = select(data, dataset | all_of(columns))
    if (!is.null(unlinked)) {
        fracs = mutate(fracs, unlinked=rowSums(select(data,
        all_of(unlinked))))
    }
    totals = rowSums(select(fracs, !dataset))
    fracs %>% mutate(across(!dataset, ~ .x / totals))
}
fractionalize(book_counts, link_codes) %>% glimpse()
```

```
Rows: 7
Columns: 5
$ dataset   <chr> "LOC-MDS", "BX-I", "BX-E", "AZ14", "AZ18", "GR-
I", "GR-E"
$ female    <dbl> 0.1717938, 0.3320289, 0.3365408, 0.2339556,
0.2414279, 0.328…
$ male      <dbl> 0.5603916, 0.4775684, 0.4889272, 0.5178783,
0.5093450, 0.487…
$ ambiguous <dbl> 0.01710506, 0.04428229, 0.04678265, 0.02262252,
0.02124007, …
$ unknown   <dbl> 0.2507096, 0.1461204, 0.1277494, 0.2255436,
0.2279871, 0.156…
```

And a helper function for plotting bar charts:

```
plot_bars = function(data, what="UNSPECIFIED") {
    tall = data %>%
```

```
        pivot_longer(!dataset, names_to="status", values_to="fraction")
    codes = c(all_codes, "unlinked")
    codes = intersect(codes, unique(tall$status))
    tall = tall %>% mutate(status=ordered(status, codes))
    ggplot(tall) +
        aes(y=dataset, x=fraction, fill=status) +
        geom_col(position=position_stack(reverse=TRUE), width=0.5) +
        geom_text(aes(label=if_else(fraction >= 0.1,
                                    sprintf("%.1f%%", fraction *
        100),
                                    "")),
                  position=position_stack(reverse=TRUE, vjust=0.5),
                  colour="white", fontface="bold") +
        scale_fill_brewer(type="qual", palette="Dark2") +
        ylab("Dataset") +
        xlab(paste("Fraction of", what)) +
        labs(fill="Author Gender")
}
```

Resolution of Books

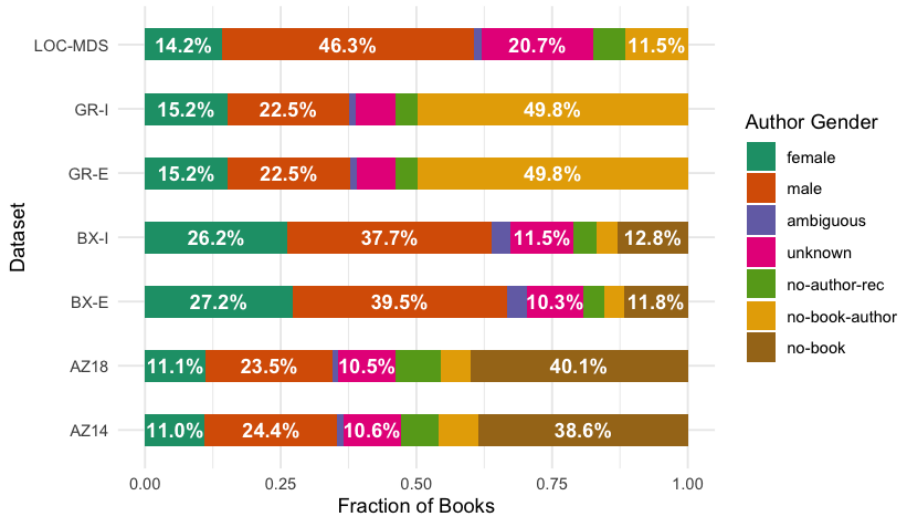
What fraction of *unique books* are resolved from each source?

```
book_counts %>% fractionalize(all_codes)
```

A tibble: 7 × 8

dataset <chr>	female <dbl>	male <dbl>	ambiguous <dbl>	unknown <dbl>	no-author-rec <dbl>	no-book-author <dbl>	no-book <dbl>
LOC-MDS	0.1420289	0.4632984	0.014141442	0.20727173	0.05854128	0.11471829	0.00000
BX-I	0.2620755	0.3769521	0.034952696	0.11533509	0.04241426	0.03984270	0.12842
BX-E	0.2715999	0.3945810	0.037755198	0.10309814	0.03840289	0.03662173	0.11794
AZ14	0.1102410	0.2440267	0.010659834	0.10627718	0.06888803	0.07439735	0.38551
AZ18	0.1114070	0.2350375	0.009801243	0.10520474	0.08405064	0.05340393	0.40109
GR-I	0.1515577	0.2248109	0.012428636	0.07196704	0.04092236	0.49831331	0.00000
GR-E	0.1522048	0.2251909	0.012478855	0.07177633	0.04078407	0.49756502	0.00000

```
book_counts %>% fractionalize(all_codes) %>% plot_bars("Books")
```

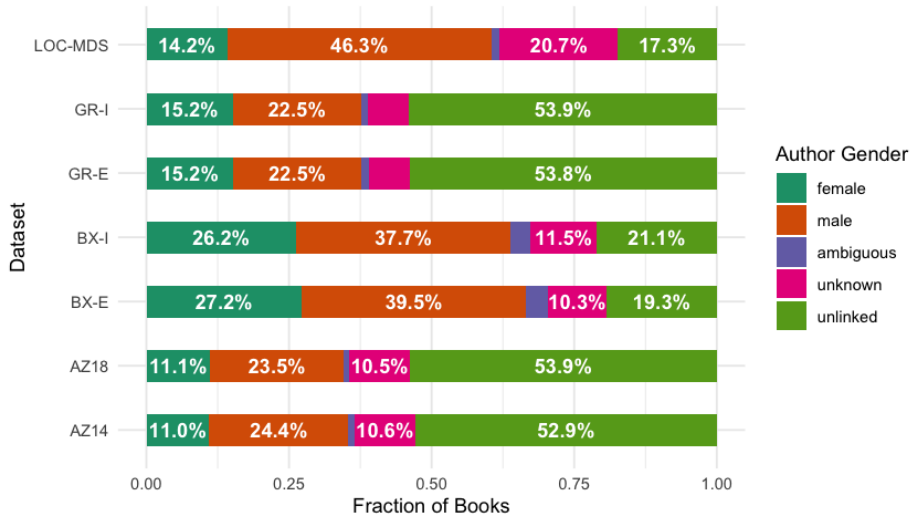



```
book_counts %>% fractionalize(link_codes, unlink_codes)
```

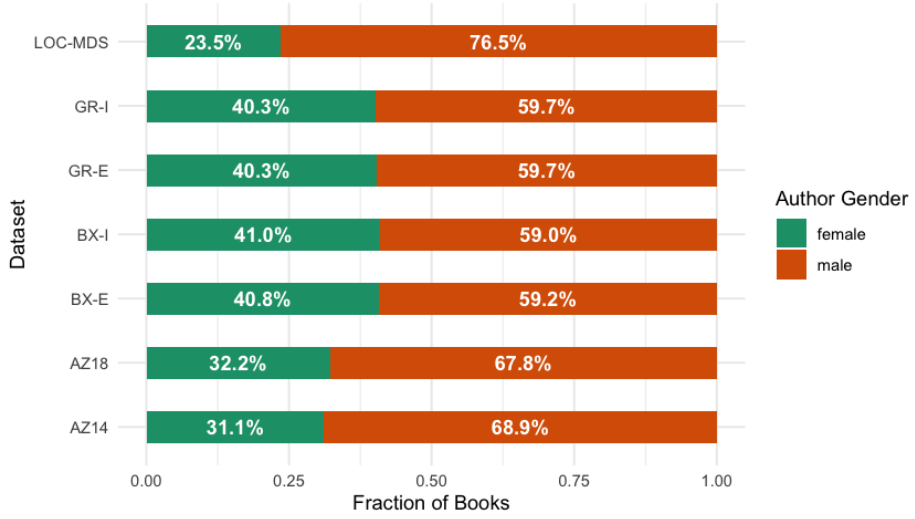
A tibble: 7 × 6

dataset <chr>	female <dbl>	male <dbl>	ambiguous <dbl>	unknown <dbl>	unlinked <dbl>
LOC-MDS	0.1420289	0.4632984	0.014141442	0.20727173	0.1732596
BX-I	0.2620755	0.3769521	0.034952696	0.11533509	0.2106846
BX-E	0.2715999	0.3945810	0.037755198	0.10309814	0.1929658
AZ14	0.1102410	0.2440267	0.010659834	0.10627718	0.5287954
AZ18	0.1114070	0.2350375	0.009801243	0.10520474	0.5385495
GR-I	0.1515577	0.2248109	0.012428636	0.07196704	0.5392357
GR-E	0.1522048	0.2251909	0.012478855	0.07177633	0.5383491

```
book_counts %>% fractionalize(link_codes, unlink_codes) %>%
        plot_bars("Books")
```




```
book_counts %>% fractionalize(c('female', 'male')) %>%
        plot_bars("Books")
```



Resolution of Ratings

What fraction of *rating actions* have each resolution result?

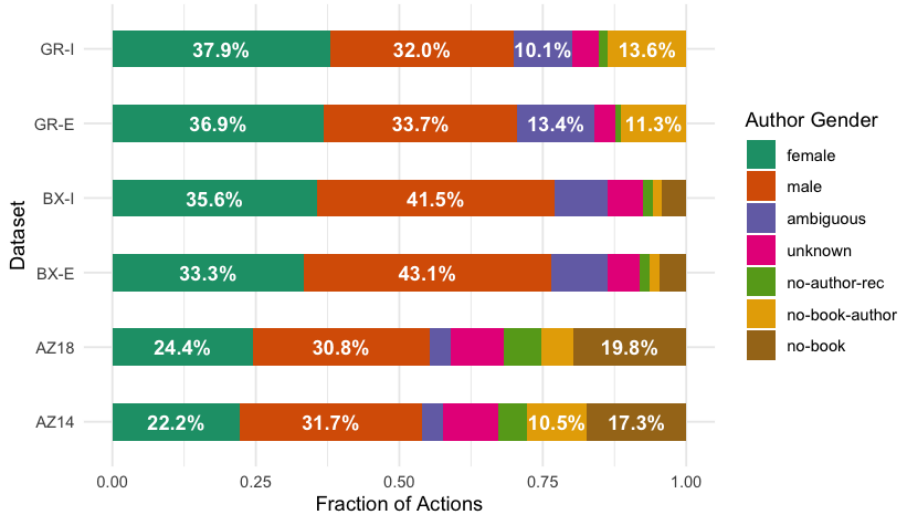
```
act_counts %>% fractionalize(all_codes)
```

A tibble: 6 × 8

dataset <chr>	female <dbl>	male <dbl>	ambiguous <dbl>	unknown <dbl>	no-author-rec <dbl>	no-book-author <dbl>	no-book <dbl>
BX-I	0.3559998	0.4151195	0.09222513	0.06150322	0.01649018	0.01674289	0.041919
BX-E	0.3332966	0.4309834	0.09786248	0.05753005	0.01670560	0.01694927	0.046672
AZ14	0.2219285	0.3168158	0.03785655	0.09618871	0.04905276	0.10519131	0.172966
AZ18	0.2443175	0.3080009	0.03641218	0.09263233	0.06538412	0.05568122	0.197571
GR-I	0.3793034	0.3202166	0.10108856	0.04687064	0.01622631	0.13629452	0.000000

dataset <chr>	female <dbl>	male <dbl>	ambiguous <dbl>	unknown <dbl>	no-author-rec <dbl>	no-book-author <dbl>	no-book <dbl>
GR-E	0.3685359	0.3372415	0.13419609	0.03621023	0.01054240	0.11327395	0.000000

```
act_counts %>% fractionalize(all_codes) %>% plot_bars("Actions")
```

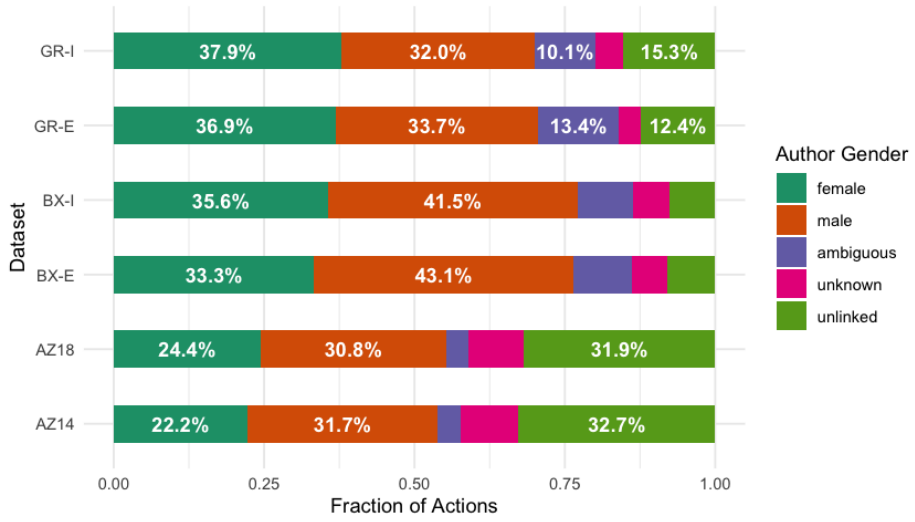



```
act_counts %>% fractionalize(link_codes, unlink_codes)
```

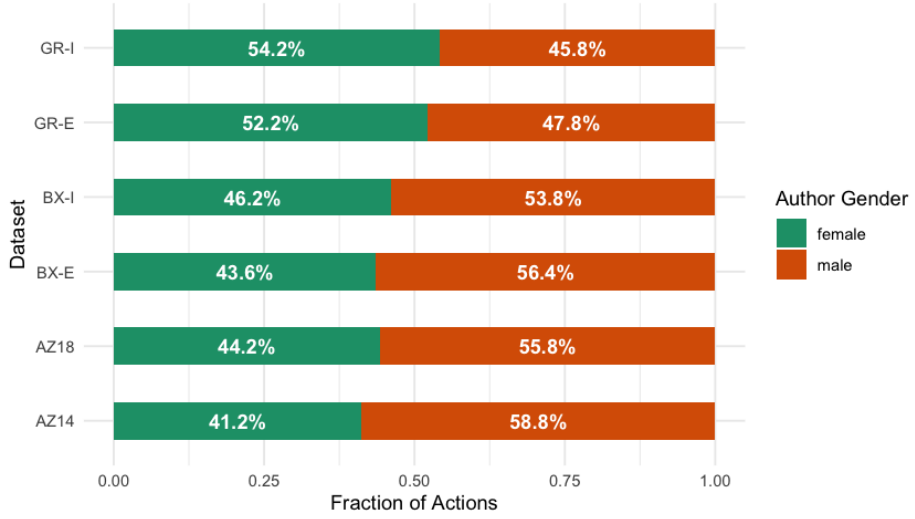
A tibble: 6 × 6

dataset <chr>	female <dbl>	male <dbl>	ambiguous <dbl>	unknown <dbl>	unlinked <dbl>
BX-I	0.3559998	0.4151195	0.09222513	0.06150322	0.07515238
BX-E	0.3332966	0.4309834	0.09786248	0.05753005	0.08032746
AZ14	0.2219285	0.3168158	0.03785655	0.09618871	0.32721043
AZ18	0.2443175	0.3080009	0.03641218	0.09263233	0.31863702
GR-I	0.3793034	0.3202166	0.10108856	0.04687064	0.15252082
GR-E	0.3685359	0.3372415	0.13419609	0.03621023	0.12381635

```
act_counts %>% fractionalize(link_codes, unlink_codes) %>%
        plot_bars("Actions")
```




```
act_counts %>% fractionalize(c('female', 'male')) %>%
        plot_bars("Actions")
```



Metrics

Finally, we're going to write coverage metrics.

```
book_linked = eval(quote(male + female + ambiguous),
        envir=book_counts)
book_coverage = book_linked / book_counts$total
book_coverage = setNames(book_coverage, book_counts$dataset)
book_coverage
```

LOC-MDS
    0.619468703489958BX-I
    0.673980271242897BX-E
    0.703936094131617AZ14
    0.364927444554598AZ18
    0.356245756601834GR-I
    0.388797287476234GR-E
    0.389874577938927

```
json = toJSON(
    as.list(book_coverage),
    auto_unbox=TRUE,
)
write_file(json, "book-coverage.json")
```