This post shows how to plot up an age profile from the Scottish census table KS102SC, which is [available online](https://www.scotlandscensus.gov.uk/ods-web/data-warehouse.html).

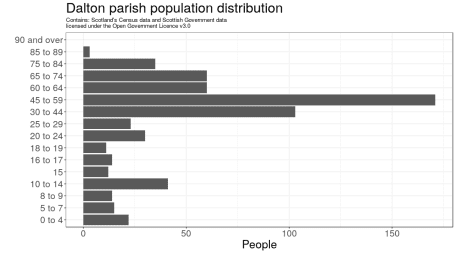
First up, let’s load our packages and read in the table. Note I’ve skipped the first few header lines and have coded – to NA. In reality – are actually 0s, so I’ve used `mutate\_all` to fix them.

library(tidyverse)

df = read\_csv("~/Downloads/temp/KS102SC.csv", skip=4, na="-") %>%

mutate\_all(funs(replace(., is.na(.), 0)))

Next we can select the parish of interest, select the columns we’re interested in, convert these to long format, and force the ordering of the ages (e.g. 8-10 should come before 10-14). I’ve piped the output of this munging into ggplot and added some styling and an all important licence statement.

[](https://scottishsnow.files.wordpress.com/2018/11/dalton.png)

df %>%

filter(X1=="Dalton") %>%

select(-X1, -`All people`, -`Mean age`, -`Median age`, -X21) %>%

gather() %>%

mutate(key = reorder(key, seq\_along(key))) %>%

ggplot(aes(key, value)) +

geom\_col() +

labs(title="Dalton parish population distribution",

subtitle="Contains: Scotland's Census data and Scottish Government data\nlicensed under the Open Government Licence v3.0",

x="",

y="People") +

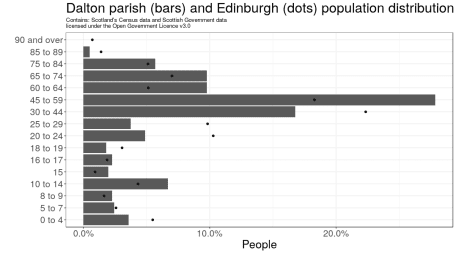
coord\_flip() +

theme\_bw() +

theme(text=element\_text(size=20),

plot.subtitle=element\_text(size=10))

It’s also of interest to compare one parish against another, so I compared Dalton against Edinburgh. Basically as before but adding an extra point layer for the visualisation. The data have now been changed to proportions of each parish so they are comparable.

[](https://scottishsnow.files.wordpress.com/2018/11/dalton_edinburgh.png)

x = df %>%

filter(X1=="Dalton" | X1=="Edinburgh") %>%

select(-`Mean age`, -`Median age`, -X21) %>%

mutate\_at(vars(-X1), funs(prop = . / `All people`)) %>%

select(-`All people\_prop`) %>%

select(X1, ends\_with("prop")) %>%

gather(key, value, -X1) %>%

separate(key, c("key", "drop"), "\_") %>%

mutate(key = reorder(key, seq\_along(key)))

x %>%

filter(X1=="Dalton") %>%

ggplot(aes(key, value)) +

geom\_col() +

geom\_point(data=filter(x, X1=="Edinburgh"), aes(key, value)) +

scale\_y\_continuous(labels=scales::percent) +

labs(title="Dalton parish (bars) and Edinburgh (dots) population distribution",

subtitle="Contains: Scotland's Census data and Scottish Government data\nlicensed under the Open Government Licence v3.0",

x="",

y="People") +

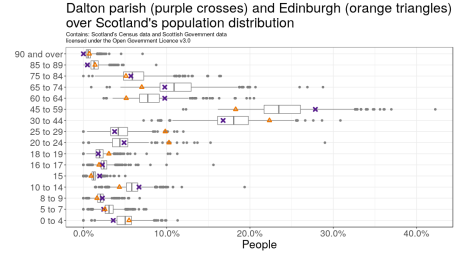
coord\_flip() +

theme\_bw() +

theme(text=element\_text(size=20),

plot.subtitle=element\_text(size=10))

Finally, we can compare distributions for the whole of Scotland against Edinburgh and Dalton using boxplots. I can imagine a beautiful plot with density polygons showing the national data, but I don’t have time to figure it out now!

[](https://scottishsnow.files.wordpress.com/2018/11/boxplots.png)

x = df %>%

select(-`Mean age`, -`Median age`, -X21) %>%

mutate\_at(vars(-X1), funs(prop = . / `All people`)) %>%

select(-`All people\_prop`) %>%

select(X1, ends\_with("prop")) %>%

gather(key, value, -X1) %>%

separate(key, c("key", "drop"), "\_") %>%

mutate(key = reorder(key, seq\_along(key)))

x %>%

filter(X1!="Scotland") %>%

ggplot(aes(key, value)) +

geom\_boxplot(colour="grey50") +

geom\_point(data=filter(x, X1=="Dalton"), aes(key, value), colour="purple4", shape=4, stroke=2, show.legend=T) +

geom\_point(data=filter(x, X1=="Edinburgh"), aes(key, value), colour="darkorange2", shape=2, stroke=1.5, show.legend=T) +

scale\_y\_continuous(labels=scales::percent) +

labs(title="Dalton parish (purple crosses) and Edinburgh (orange triangles)\nover Scotland's population distribution",

subtitle="Contains: Scotland's Census data and Scottish Government data\nlicensed under the Open Government Licence v3.0",

x="",

y="People") +

coord\_flip() +

theme\_bw() +

theme(text=element\_text(size=20),

plot.subtitle=element\_text(size=10))