For Tina

Import

```
babies <- readxl::read_xlsx("PICC-Data-1.xlsx", 1)
pokes <- readxl::read_xlsx("PICC-Data-1.xlsx", 2)
```

Tidy

How does it make sense that there can be several conflicting entries per study participant in your demographic data? Let's just keep the first entry for simplicity. In real life you should figure out why your data is so awful.

```
babies <- babies %>%
  distinct(`Study Participant #`, .keep_all=TRUE) %>%
  arrange(`Study Participant #`)
```

Now that our data on the babies is ready, let's join it to our pokes data.

```
pokes <- pokes %>%
  left_join(babies, by="Study Participant #")
```

We only have 130 pokes in total, but upwards of 10 features. Since this is a human clinical trial with so many uncontrolled variables, it's unlikely we have enough data to run such a complex model. For comparison, stringent mice experiments usually use at least **10 subjects per group**. Before we investigate how many numbers in each group for some of these features, let's drop the dates/times as I assume the protocol hasn't changed over time, and drop the comments/consent/participant#.

```
pokes <- pokes %>%
  select(-matches("[Dd]ate"),
         -matches("time"),
         -matches("[Cc]omplication"),
         -matches("[Cc]onsent"),
         -matches("participant #"))
```

Categorical Variables

How are our numbers for site of insertion columns? I see a lot of NAs.

```
pokes %>% select(contains("site")) %>%
  table(useNA="always")
```

```
##                                  Site of Insertion
## Site of Insertion: Right or Left?  0  1  2  3  4  5  6 <NA>
##                                0     5  2 13 10  9  1  0    0
##                                1     4  2 10  9  8  2  1    1
##                                <NA>  7  4  7 12  7  4  1   11
```

```
pokes <- pokes %>% select(-`Site of Insertion`)
```

Let's drop the specific site of insertion while keeping the right/left split.

Let's investigate the type of cather and brand name of PICC.

```
pokes %>%
  select(matches("Type of Catheter"), matches("Brand name"))%>%
  table(useNA="always")
```

```
##                       Brand Name of PICC
## Type of Catheter Used  ?  0  1  2 <NA>
##                   0     1 47 24  0    1
##                   1     1  6  3  2    0
##                   <NA>  0 33 11  0    1
```

```
pokes <- pokes %>% select(-matches("Type of Catheter"))
```

Let's drop the type of catheter. There's no point in comparing type 0 versus NA. Also, drawing from my limited domain knowledge of poking needles, the catheter probably doesn't have much to do with the success.

Do we have a lot of data on tip position and if its trimmed?

```
pokes %>%
  select(matches("PICC ")) %>%
  table(useNA="always")
```

```
##                   PICC Line Trimmed?
## PICC Tip Position  0  1 <NA>
##              0     3  4    2
##              1    16 18    1
##              <NA> 19 27   40
```

```
pokes <- pokes %>% select(-matches("tip position"))
```

Let's drop the tip position. Since we are keeping whether the tip wa trimmed, let's investigate *how* the tip was trimmed.

```
pokes %>%
  select(matches("trim")) %>%
  table(useNA="always")
```

```
##                    If trimmed, how?
## PICC Line Trimmed?  0  1  3 <NA>
##                0    34  1  0    3
##                1     0 23 24    2
##                <NA>  0  0  0   43
```

Okay cool, it seems like we have three levels for "trim". 0 for no trim, and 1 or 3. Despite being numbers, these are factors and not a continuous input, as the "distance" between not trimmed versus a type 1 trim is not twice as long as the distance between a type 1 trim and a type 3 trim.

We can also drop the "if trimmed" column.

```
pokes <- pokes %>%
  mutate(trim = factor(`If trimmed, how?`)) %>%
  select(-matches("Line Trimmed?"), -matches("If trimmed, how?"))
```

There are so many NAs in dressing. What even is that anyways? Drop it.

```
pokes$`Dressing Used` %>%
  table(useNA="always")
```

```
## .
##    0    1    2 <NA>
##   18   29    1   82
```

```
pokes <- pokes %>%
  select(-`Dressing Used`)
```

We must drop "Total Number of Pokes", since that can't possibly be a predictor, but "attempt number" is fine.

```
pokes <- pokes %>%
  select(-matches("total number"))
pokes %>%
  select(matches("attempt number")) %>%
  table(useNA="always")
```

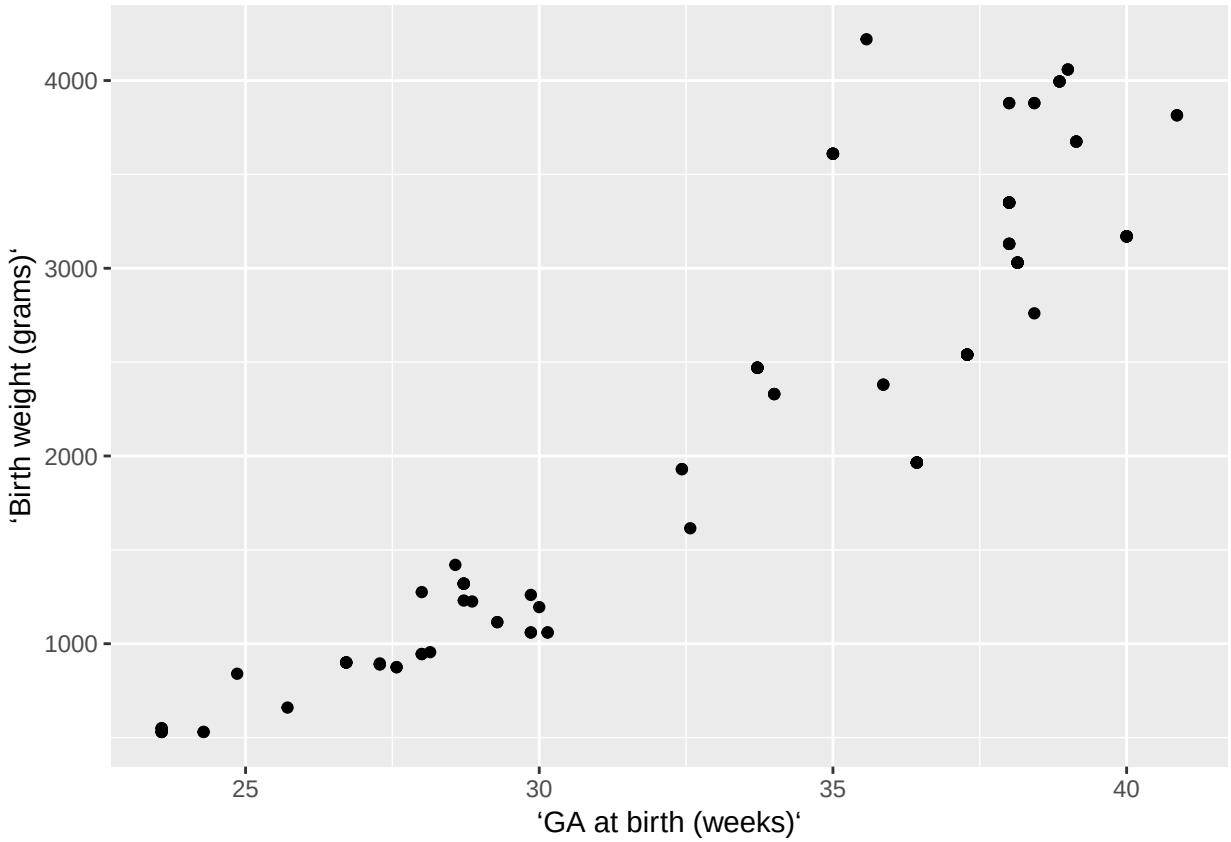
```
## .
##    1    2    3    4 <NA>
##   68   37   20    5    0
```

Continuous Variables

Now, let's start looking at our continuous variables. GA and weight are probably correlated and redundant.

```
pokes %>%
  ggplot() +
  geom_point(aes(x=`GA at birth (weeks)`, y=`Birth weight (grams)`))
```

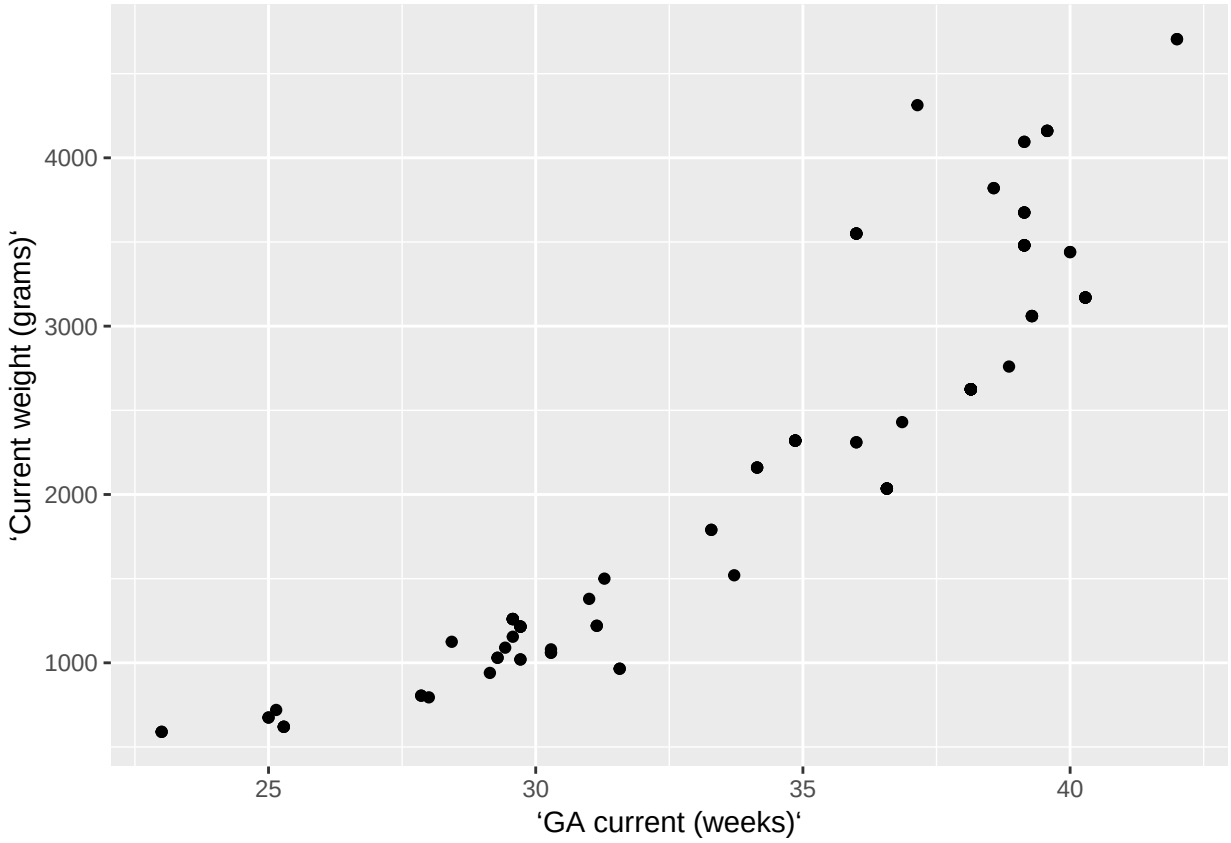
```
## Warning: Removed 4 rows containing missing values (geom_point).
```




```
pokes %>%
  ggplot() +
```

```
  geom_point(aes(x=`GA current (weeks)`, y=`Current weight (grams)`))
```

```
## Warning: Removed 15 rows containing missing values (geom_point).
```



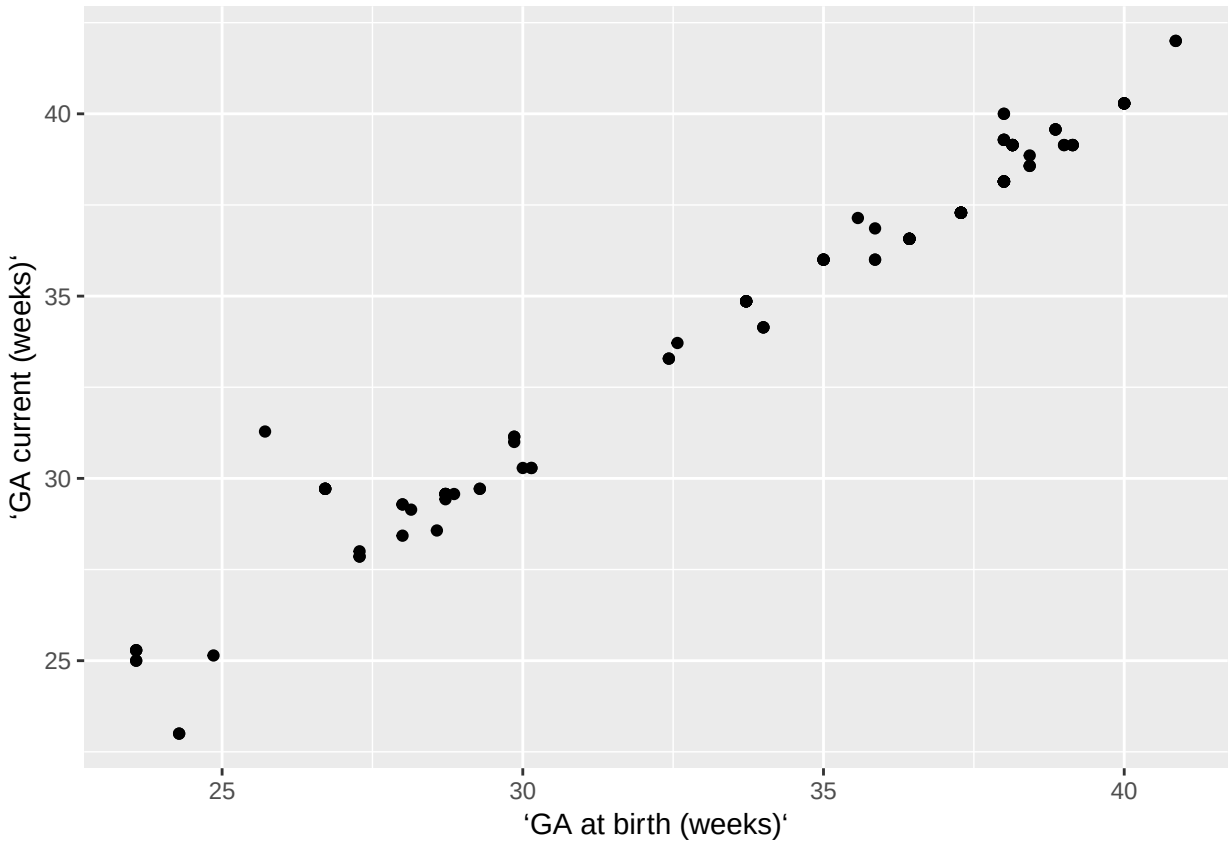
Due to lack of domain knowledge, I really don't know which to keep, but I'll just keep gestation age since we'd have less NAs.

```
pokes <- pokes %>%
  select(-matches("weight"))
```

Also, do the pokes happen after a set amount of time after birth? If so, gestation age at birth and current would be redundant variables.

```
pokes %>%
  ggplot() +
  geom_point(aes(x=`GA at birth (weeks)`, y=`GA current (weeks)`))
```

```
## Warning: Removed 5 rows containing missing values (geom_point).
```



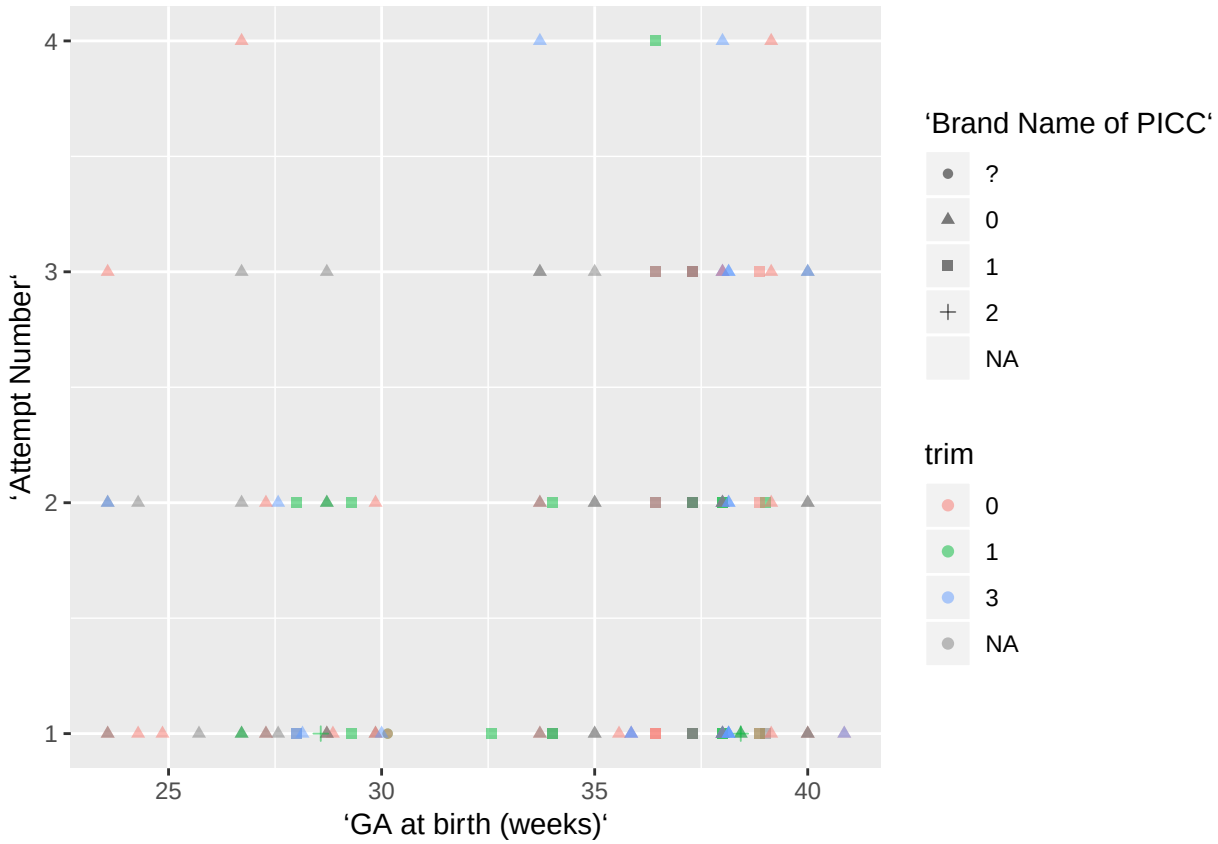
As suspected. Lets drop current gestation age so we have less NAs.

```
pokes <- pokes %>% select(-matches("ga current"))
```

Visualise Predictors

```
pokes %>%
  ggplot() +
  geom_point(aes(x=`GA at birth (weeks)`, y=`Attempt Number`, shape=`Brand Name of PICC`, color=trim ),
```

```
## Warning: Removed 5 rows containing missing values (geom_point).
```


'Attempt Number'
4
3
2
1
25
30
35
40
'GA at birth (weeks)'
'Brand Name of PICC'
?
0
1
2
NA
trim
0
1
3
NA