Model measurements

Andy Baxter

Results from Monte-carlo simuation

```
cv_class_log_no_hh <- readRDS("../output/cv_class_log_no_hh.rds")
cv_class_xg_no_hh <- readRDS("../output/cv_class_xg_no_hh.rds")
```

Metrics

```
met_log <- collect_metrics(cv_class_log_no_hh)
met_xg <- collect_metrics(cv_class_xg_no_hh)

sens_log <- met_log$mean[met_log$.metric == "sens"]
spec_log <- met_log$mean[met_log$.metric == "spec"]

sens_xg <- met_xg$mean[met_xg$.metric == "sens"]
spec_xg <- met_xg$mean[met_xg$.metric == "spec"]
```

The logistic regression model has a sensitivity of 59.7% and a specificity of 93.5%. Using this prediction model, 40.3% of UC-eligible participants will be absent from the analysis; 6.5% of non-UC-eligible participants will be included in the analysis.

The boost tree model has a sensitivity of 63.7% and a specificity of 93.1%. Using this prediction model, 36.3% of UC-eligible participants will be absent from the analysis; 6.9% of non-UC-eligible participants will be included in the analysis.

Other cutoff points

Decreasing threshold to 30% probability of receiving UC:

```r
cv_class_log_no_hh |>
  select(-splits,-.metrics,-.notes) |>
  mutate(spec = map(.predictions, function(df) {
    df |>
      transmute(pred_class = fct_rev(factor(if_else(
        .pred_Yes > 0.3, "Yes", "No"
      ))),
      uc_receipt = uc_receipt) |>
      (\(x) {
      bind_rows(spec(x, uc_receipt, pred_class),
                sens(x, uc_receipt, pred_class))

      })()

  })) |>
  unnest(spec) |>
  group_by(.metric) |>
  summarise(estimate = mean(.estimate))
```

```
# A tibble: 2 x 2
  .metric estimate
  <chr>      <dbl>
1 sens       0.760
2 spec       0.841
```

```r
cv_class_xg_no_hh |>
  select(-splits,-.metrics,-.notes) |>
  mutate(spec = map(.predictions, function(df) {
    df |>
      transmute(pred_class = fct_rev(factor(if_else(
        .pred_Yes > 0.3, "Yes", "No"
      ))),
      uc_receipt = uc_receipt) |>
      (\(x) {
      bind_rows(spec(x, uc_receipt, pred_class),
                sens(x, uc_receipt, pred_class))

      })()

  })) |>
```

```
  unnest(spec) |>
  group_by(.metric) |>
  summarise(estimate = mean(.estimate))
```

```
# A tibble: 2 x 2
  .metric estimate
  <chr>      <dbl>
1 sens       0.770
2 spec       0.860
```