

LHR: Amsterdam Bias Analysis

Justin Braun

2024-10-13

Load Libraries

```
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

library(tidyr)
library(ggplot2)
library(openxlsx)
library(stringr)
library(friendlyeval)
```

Load Data

```
cms_raw <- read.xlsx('../input/Results_LHR/Output/20240308_CMs_LHR_SlimmeCheck.xlsx')
feature_counts_raw <- read.xlsx('../input/Results_LHR/Output/20240308_Important_Features_Counts.xlsx')
```

Preprocessing

```
### Confusion Matrices ###
summary(cms_raw)
```

```
##   Dataset           Model           Feature           Feature_Value
## Length:480         Length:480       Length:480       Length:480
## Class :character   Class :character Class :character Class :character
```

```
## Mode :character Mode :character Mode :character Mode :character
##
##
##
## Metric Value
## Length:480 Min. : 0.0
## Class :character 1st Qu.: 32.0
## Mode :character Median : 83.0
## Mean : 159.9
## 3rd Qu.: 202.2
## Max. :1024.0
```

```
print(unique(cms_raw$Dataset))
```

```
## [1] "Pilot" "Prepilot" "TrainingTrain" "TrainingTest"
```

```
print(unique(cms_raw$Model))
```

```
## [1] "BR" "AR"
```

```
print(unique(cms_raw$Feature))
```

```
## [1] "geslacht" "Leeftijd<30" "Leeftijd<40" "Leeftijd<50"
## [5] "IsNederlands" "IsWesters" "IsFulltimeParent" "IsParttimeParent"
```

```
cms_raw <- cms_raw %>%
  #0s indicate small sample sizes but are unlikely to be correct
  mutate(Value = ifelse(Value == 0, NA, Value))

#combine train and test since the original split is not actually recreated
#TODO: current approach results in NAs if either Train or Test is NA/0,
#I could just go with using data from the split that is sufficiently large in those cases
cms_train <- cms_raw %>%
  filter(Dataset == 'TrainingTrain') %>%
  rename(Value_Train = Value) %>%
  select(-Dataset)

cms_test <- cms_raw %>%
  filter(Dataset == 'TrainingTest') %>%
  rename(Value_Test = Value) %>%
  select(-Dataset)

cms_train_test <- cms_train %>%
  left_join(cms_test, by = c('Model', 'Feature', 'Feature_Value', 'Metric')) %>%
  mutate(Value = Value_Train + Value_Test,
         Dataset = 'TrainTest') %>%
  select(-Value_Train, -Value_Test)

cms_wide <- cms_raw %>%
  filter(!(Dataset %in% c('TrainingTrain', 'TrainingTest'))) %>%
  bind_rows(cms_train_test) %>%
```

```

group_by(Feature, Feature_Value, Dataset, Model) %>%
mutate(Share = (Value/sum(Value)) * 100,
       group_size = sum(Value)) %>%
ungroup() %>%
dplyr::select(-Value) %>%
pivot_wider(names_from = Metric, values_from = Share) %>%
mutate(TOTAL = TN+FP+TP+FN,
       ACT_N = FP + TN,
       ACT_P = FN + TP,
       PRED_P = FP + TP,
       PRED_N = FN + TN,
       FPR = (FP/ACT_N) * 100,
       PPV = (TP/PRED_P) * 100,
       FDR = (FP/PRED_P) * 100,
       TPR = (TP/ACT_P) * 100,
       STAT_PAR = PRED_P,
       ERROR = FP+FN) %>%
mutate(Feature_EN = case_when(Feature == 'geslacht' ~ 'gender',
                             Feature == 'Leeftijd<30' ~ 'Age < 30',
                             Feature == 'Leeftijd<40' ~ 'Age < 40',
                             Feature == 'Leeftijd<50' ~ 'Age < 50',
                             Feature == 'IsNederlands' ~ 'Dutch',
                             Feature == 'IsWesters' ~ 'Western',
                             .default = Feature),
       Feature_Value_EN = case_when(Feature_Value == 'V' ~ 'F',
                                    Feature == 'Leeftijd<30' & Feature_Value == 1 ~ 'below 30',
                                    Feature == 'Leeftijd<30' & Feature_Value == 0 ~ 'above 30',
                                    Feature == 'Leeftijd<40' & Feature_Value == 1 ~ 'below 40',
                                    Feature == 'Leeftijd<40' & Feature_Value == 0 ~ 'above 40',
                                    Feature == 'Leeftijd<50' & Feature_Value == 1 ~ 'below 50',
                                    Feature == 'Leeftijd<50' & Feature_Value == 0 ~ 'above 50',
                                    Feature == 'IsNederlands' & Feature_Value == 1 ~ 'Dutch',
                                    Feature == 'IsNederlands' & Feature_Value == 0 ~ 'Not Dutch',
                                    Feature == 'IsWesters' & Feature_Value == 1 ~ 'Western',
                                    Feature == 'IsWesters' & Feature_Value == 0 ~ 'Not Western',
                                    Feature == 'IsFulltimeParent' & Feature_Value == 1 ~ 'Full-time p',
                                    Feature == 'IsFulltimeParent' & Feature_Value == 0 ~ 'Not full-tin',
                                    Feature == 'IsParttimeParent' & Feature_Value == 1 ~ 'Part-time p',
                                    Feature == 'IsParttimeParent' & Feature_Value == 0 ~ 'Not part-tin',
                                    .default = Feature_Value),
       is_privileged_group = case_when(Feature_Value == 'V' ~ 0,
                                       Feature_Value == 'M' ~ 1,
                                       Feature == 'Leeftijd<30' & Feature_Value == 1 ~ 1,
                                       Feature == 'Leeftijd<30' & Feature_Value == 0 ~ 0,
                                       Feature == 'Leeftijd<40' & Feature_Value == 1 ~ 1,
                                       Feature == 'Leeftijd<40' & Feature_Value == 0 ~ 0,
                                       Feature == 'Leeftijd<50' & Feature_Value == 1 ~ 1,
                                       Feature == 'Leeftijd<50' & Feature_Value == 0 ~ 0,
                                       Feature == 'IsNederlands' & Feature_Value == 1 ~ 1,
                                       Feature == 'IsNederlands' & Feature_Value == 0 ~ 0,
                                       Feature == 'IsWesters' & Feature_Value == 1 ~ 1,
                                       Feature == 'IsWesters' & Feature_Value == 0 ~ 0,
                                       Feature == 'IsFulltimeParent' & Feature_Value == 1 ~ 0,

```

```

        Feature == 'IsFulltimeParent' & Feature_Value == 0 ~ 1,
        Feature == 'IsParttimeParent' & Feature_Value == 1 ~ 0,
        Feature == 'IsParttimeParent' & Feature_Value == 0 ~ 1,
        .default = NA),

  stage = paste0(Dataset, '/', Model),
  Model_EN = case_when(Model == 'BR' ~ 'Before Reweighing',
                        Model == 'AR' ~ 'After Reweighing'))

write.csv(cms_wide, '../output/cms_wide.csv')

cms_long <- cms_wide %>%
  pivot_longer(cols = c("TN", "FP", "FN", "TP", "TOTAL", "ACT_N", "ACT_P", "PRED_P", "PRED_N", "FPR", "I"),
               names_to = 'Metric', values_to = 'Value')

write.csv(cms_long, '../output/cms_long.csv')

### Feature Importance ###
feature_counts <- feature_counts_raw %>%
  mutate(Count = ifelse(Count == 0, NA, Count),
         Feature_EN = case_when(Feature == 'geslacht' ~ 'gender',
                                Feature == 'Leeftijd<30' ~ 'Age < 30',
                                Feature == 'Leeftijd<40' ~ 'Age < 40',
                                Feature == 'Leeftijd<50' ~ 'Age < 50',
                                Feature == 'IsNederlands' ~ 'Dutch',
                                Feature == 'IsWesters' ~ 'Western',
                                .default = Feature),
         Feature_Value_EN = case_when(Feature == 'geslacht' & Value == 1 ~ 'F', #not sure about gender
                                       Feature == 'geslacht' & Value == 0 ~ 'M',
                                       Feature == 'Leeftijd<30' & Value == 1 ~ 'below 30',
                                       Feature == 'Leeftijd<30' & Value == 0 ~ 'above 30',
                                       Feature == 'Leeftijd<40' & Value == 1 ~ 'below 40',
                                       Feature == 'Leeftijd<40' & Value == 0 ~ 'above 40',
                                       Feature == 'Leeftijd<50' & Value == 1 ~ 'below 50',
                                       Feature == 'Leeftijd<50' & Value == 0 ~ 'above 50',
                                       Feature == 'IsNederlands' & Value == 1 ~ 'Dutch',
                                       Feature == 'IsNederlands' & Value == 0 ~ 'Not Dutch',
                                       Feature == 'IsWesters' & Value == 1 ~ 'Western',
                                       Feature == 'IsWesters' & Value == 0 ~ 'Not Western',
                                       Feature == 'IsFulltimeParent' & Value == 1 ~ 'Full-time parent',
                                       Feature == 'IsFulltimeParent' & Value == 0 ~ 'Not full-time parent',
                                       Feature == 'IsParttimeParent' & Value == 1 ~ 'Part-time parent',
                                       Feature == 'IsParttimeParent' & Value == 0 ~ 'Not part-time parent',
                                       .default = as.character(Value))) %>%

  group_by(Feature_EN, Feature_Value_EN, dataset) %>%
  mutate(share = (Count/sum(Count, na.rm = T)) * 100) %>% #note that I remove NAs which are presumably
  ungroup()

```

RQ 1 Results from the perspective of the city

This set of graph is meant to illustrate how the city saw the development of its model: 1. First it evaluated its model before reweighing (BR) on the test data (TrainTest) 2. Then, it built the prepilot dataset and realized substantial bias when evaluating it against BR 3. Next, it reweighed the model, seeing improvements on the prepilot set. 4. Finally, it evaluated the model against the Pilot data.

The city itself focused pretty exclusively on the share of FPs (FP widget in the graphs) (TODO: check that this is correct), but to show the tradeoffs involved in the process, I report several additional fairness metrics

- * FP: false positive share: $FP/TOTAL \rightarrow$ intuition share of people who are wrongly flagged; goal is to get it as low as possible
- * FPR: false positive rate: $FP/(FP+TN) \rightarrow$ intuition: share of people who haven't done anything wrong that are flagged; goal is to get it as low as possible
- * One_minus_PPV: positive predictive rate: $TP/(TP+FP) \rightarrow$ intuition: share of people flagged who have done something wrong; goal is to get it as high as possible
- * PRED_P: predictive parity: $TP+FP/TOTAL \rightarrow$ intuition: share of people flagged; goal is to get it as close as possible to the actual share of people who have done something wrong which is often an unknown quantity

```
cms_city_perspective <- cms_long %>%
  filter(stage %in% c('Prepilot/BR', 'Prepilot/AR'),
         Metric %in% c('STAT_PAR', 'FDR', 'FP'),
         Feature_EN %in% c('gender', 'Age < 30', 'Age < 50', 'Dutch', 'IsFulltimeParent', 'Western')) %>%
  mutate(order = case_when(stage == 'Prepilot/BR' ~ 1,
                           stage == 'Prepilot/AR' ~ 2,
                           .default = NA))

for(characteristic in unique(cms_city_perspective$Feature_EN)){
  cms_char <- cms_city_perspective %>%
    filter(Feature_EN == characteristic)

  p1 <- ggplot(cms_char, aes(x = reorder(Model_EN, order), y = Value, fill = Feature_Value_EN))+
    geom_bar(stat = 'identity', position = position_dodge())+
    facet_grid(Metric ~., scales = "free_y")+
    labs(x = 'Dataset/Model',
         title = paste0(characteristic, ': Development of Fairness Metrics from the Citys Perspective')+
         fill = characteristic)+
    theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))

  print(p1)
  ggsave(paste0('../output/rq1_p1_', characteristic, '.png'), plot = p1, width = 10, height = 8)

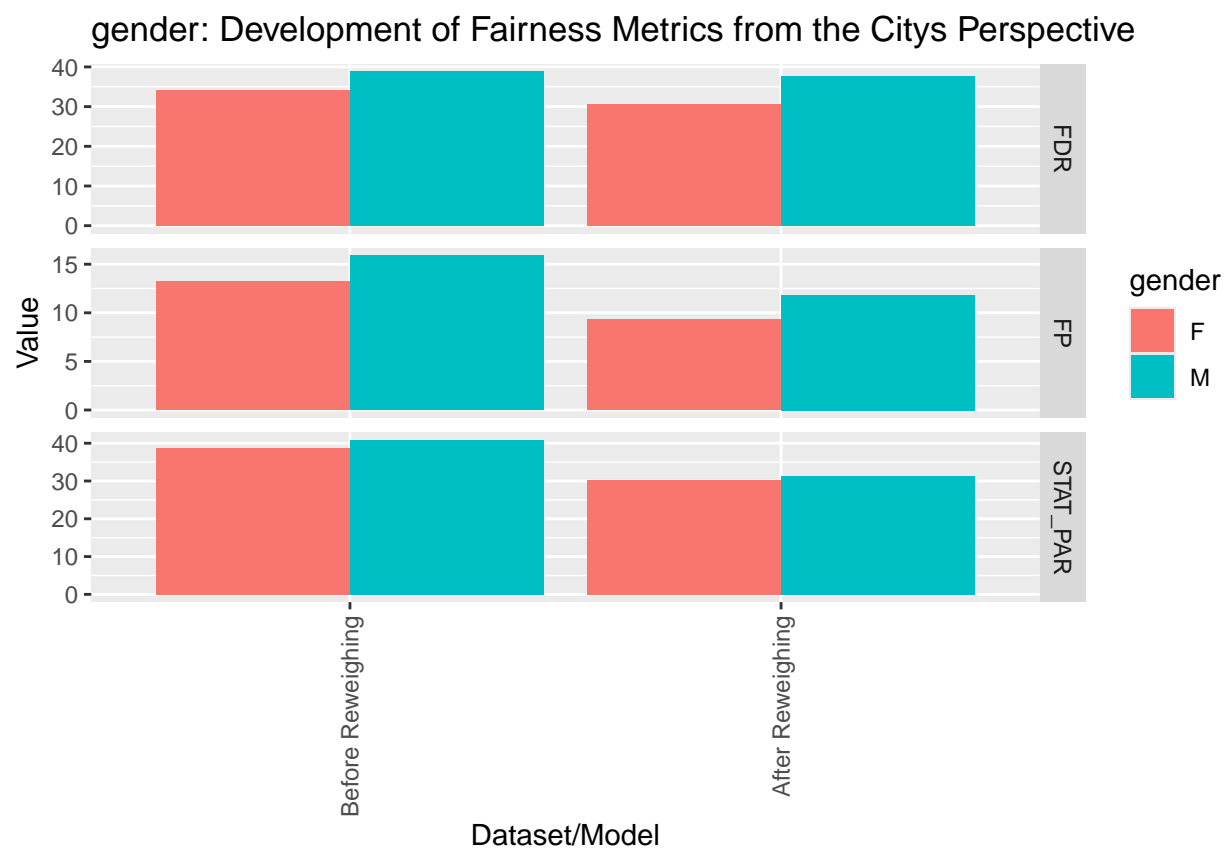
  feature_val_privileged <- unique(cms_char[cms_char$is_privileged_group == 1,]$Feature_Value_EN)[1]
  feature_val_unprivileged <- unique(cms_char[cms_char$is_privileged_group == 0,]$Feature_Value_EN)[1]

  cms_char_diff <- cms_char %>%
    dplyr::select(-Feature_Value, -group_size, -is_privileged_group) %>%
    pivot_wider(names_from = 'Feature_Value_EN', values_from = 'Value') %>%
    #using Amsterdam's difference op here, though not sure the ref cat is always the same
    mutate(Diff = 100*(.data[[feature_val_unprivileged]] - .data[[feature_val_privileged]])/.data[[feature_val_privileged]])

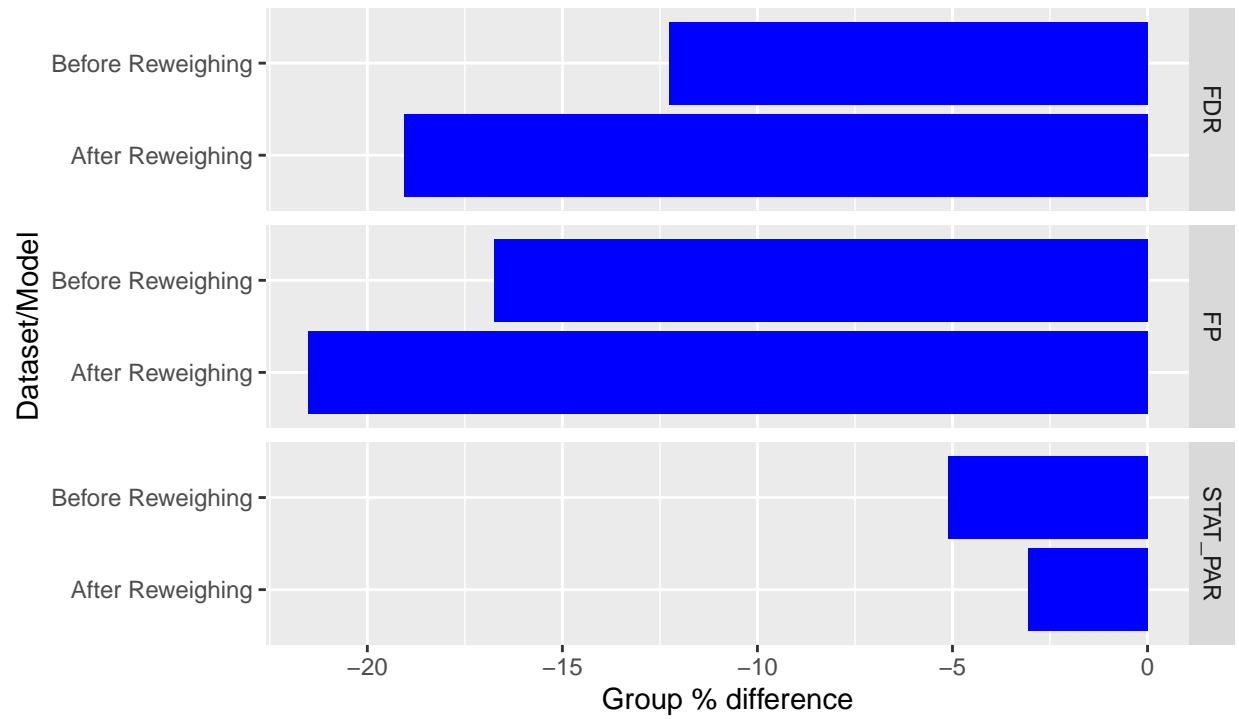
  #TODO: add difference as operationalized by the city reports
  p2 <- ggplot(cms_char_diff, aes(x = reorder(Model_EN, -order), y = Diff))+
    geom_bar(stat = 'identity', position = position_dodge(), fill = 'blue') +
    facet_grid(Metric ~., scales = "free_y")+
    labs(x = 'Dataset/Model',
         y = 'Group % difference',
         title = paste0(characteristic, ': Development of Fairness \nMetrics from the Citys Perspective')+
         subtitle = paste0(feature_val_unprivileged, ' - ', feature_val_privileged))+
    coord_flip()

  print(p2)
```

```
ggsave(paste0('../output/rq1_p2_', characteristic, '.png'), plot = p2, width = 10, height = 8)
}
```



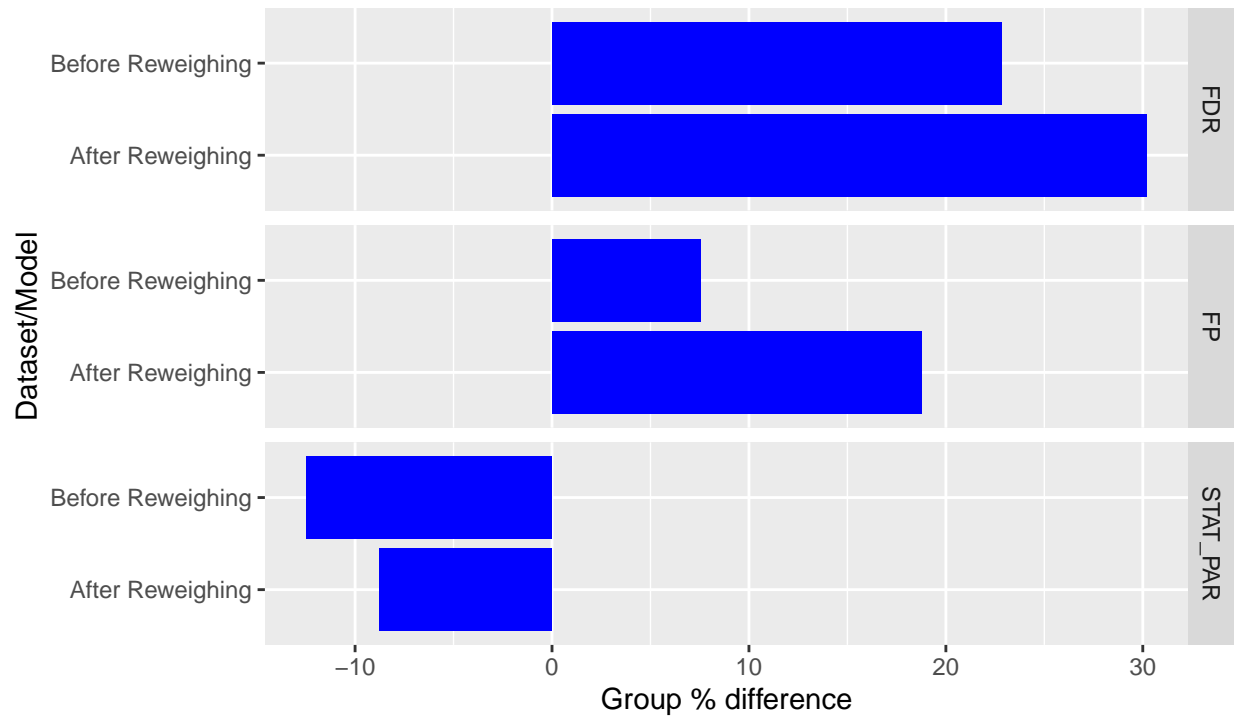
gender: Development of Fairness
Metrics from the Citys Perspective
F – M



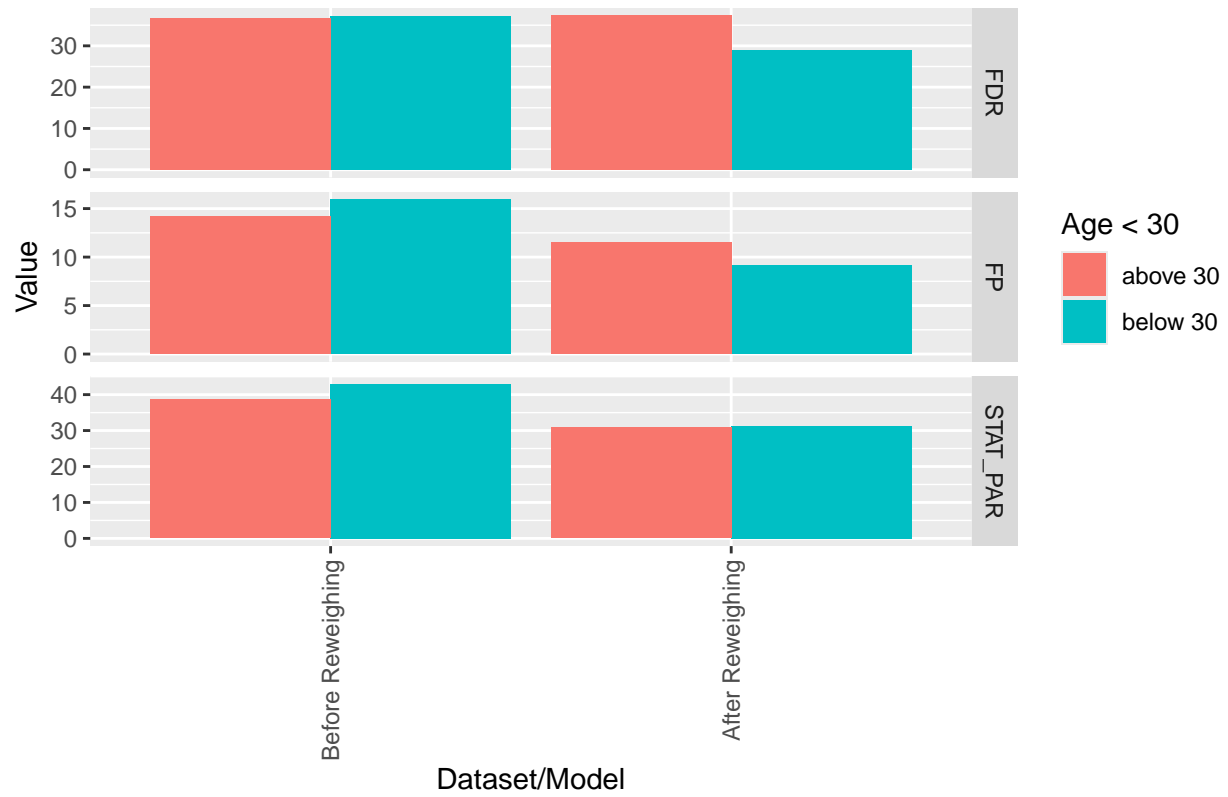
IsFulltimeParent: Development of Fairness Metrics from the City's Perspective



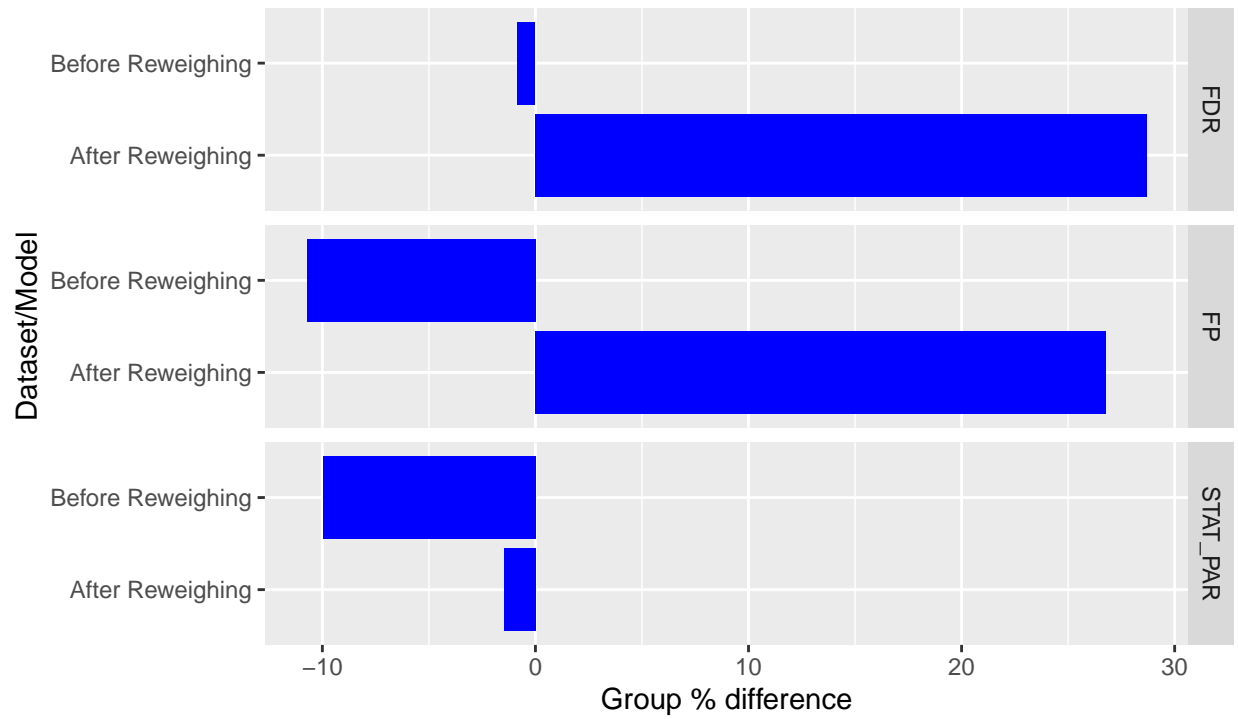
IsFulltimeParent: Development of Fairness Metrics from the Citys Perspective Full-time parent – Not full-time parent



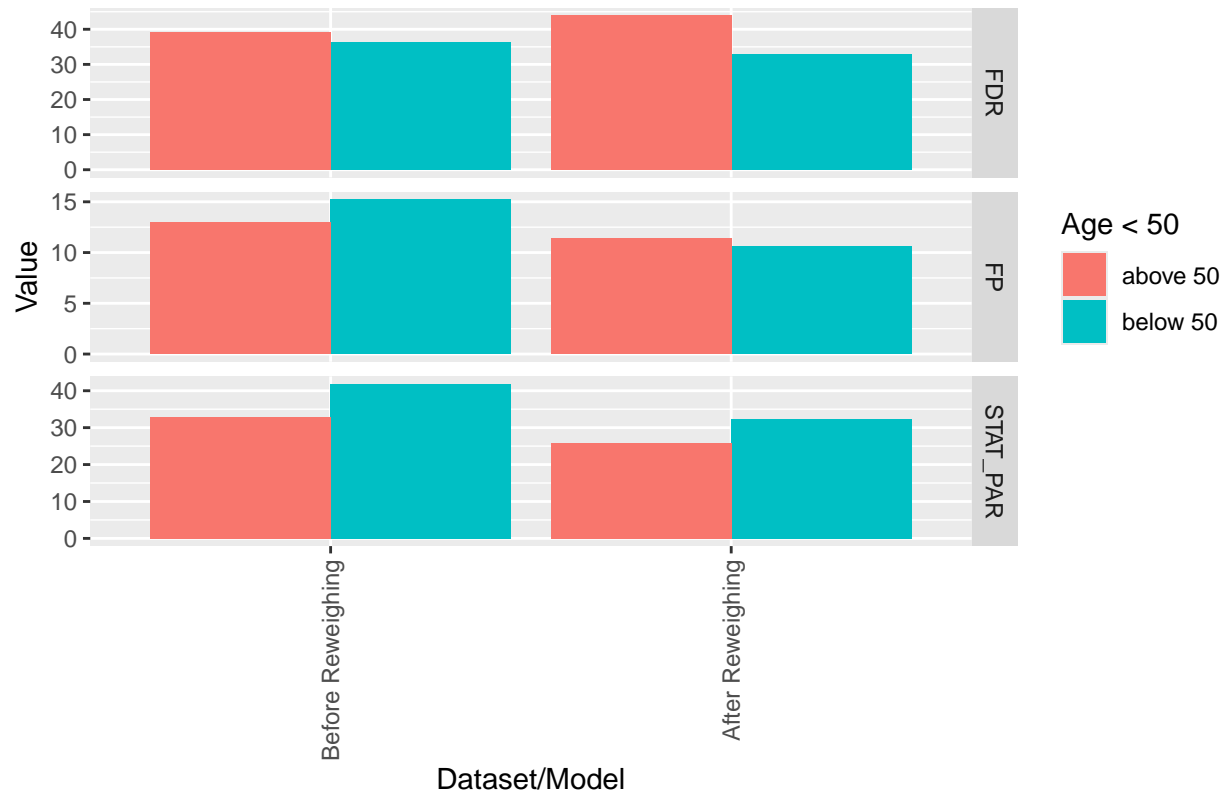
Age < 30: Development of Fairness Metrics from the City's Perspective



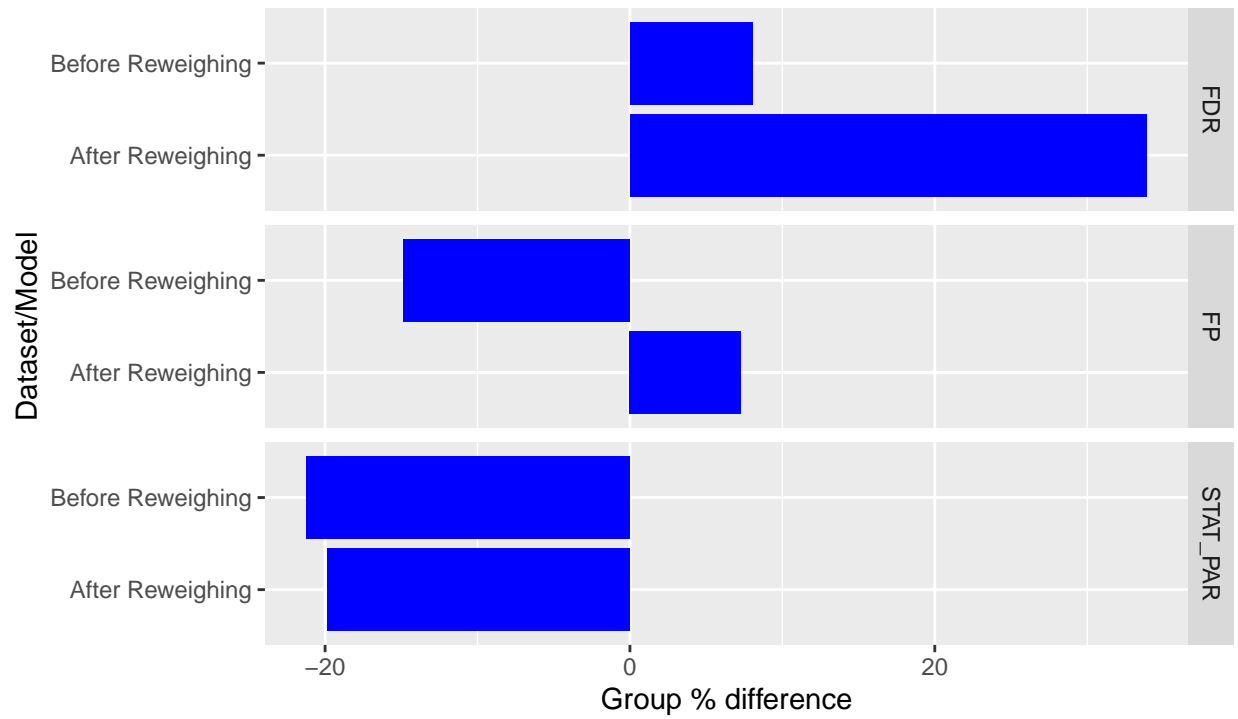
Age < 30: Development of Fairness Metrics from the Citys Perspective above 30 – below 30



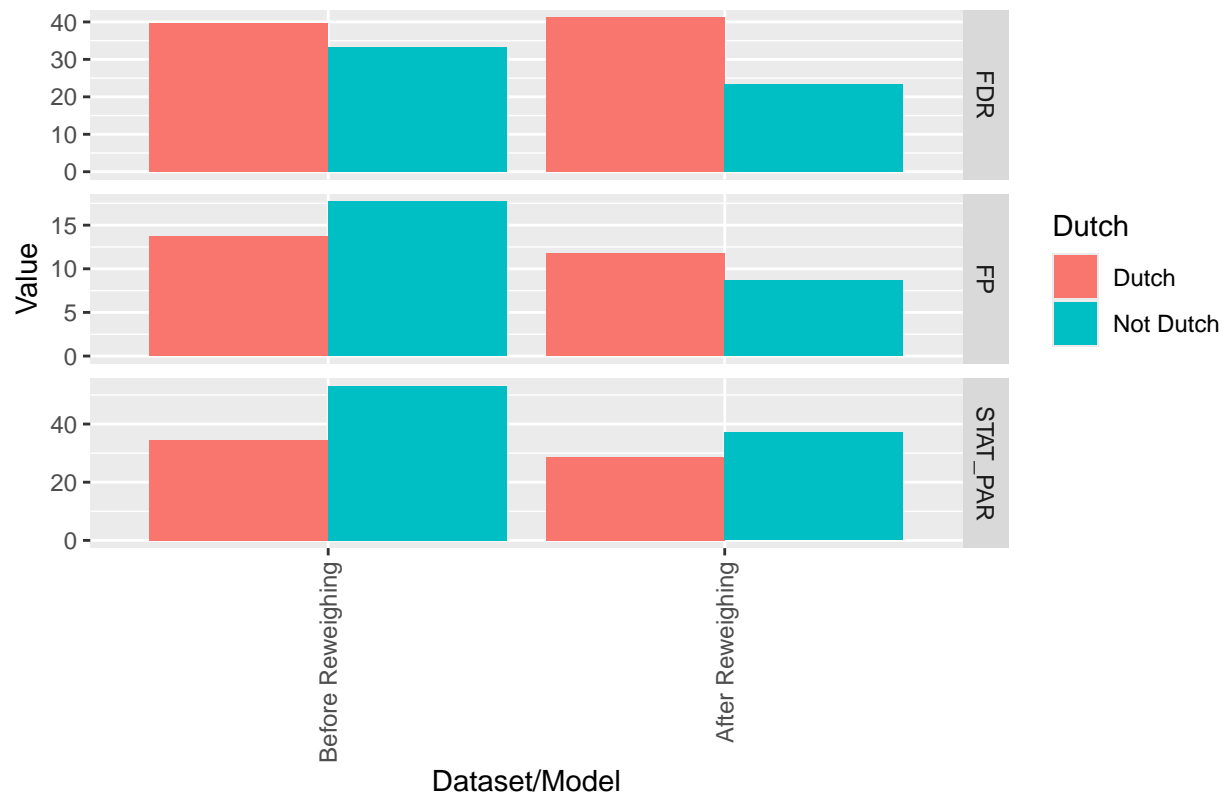
Age < 50: Development of Fairness Metrics from the City's Perspective



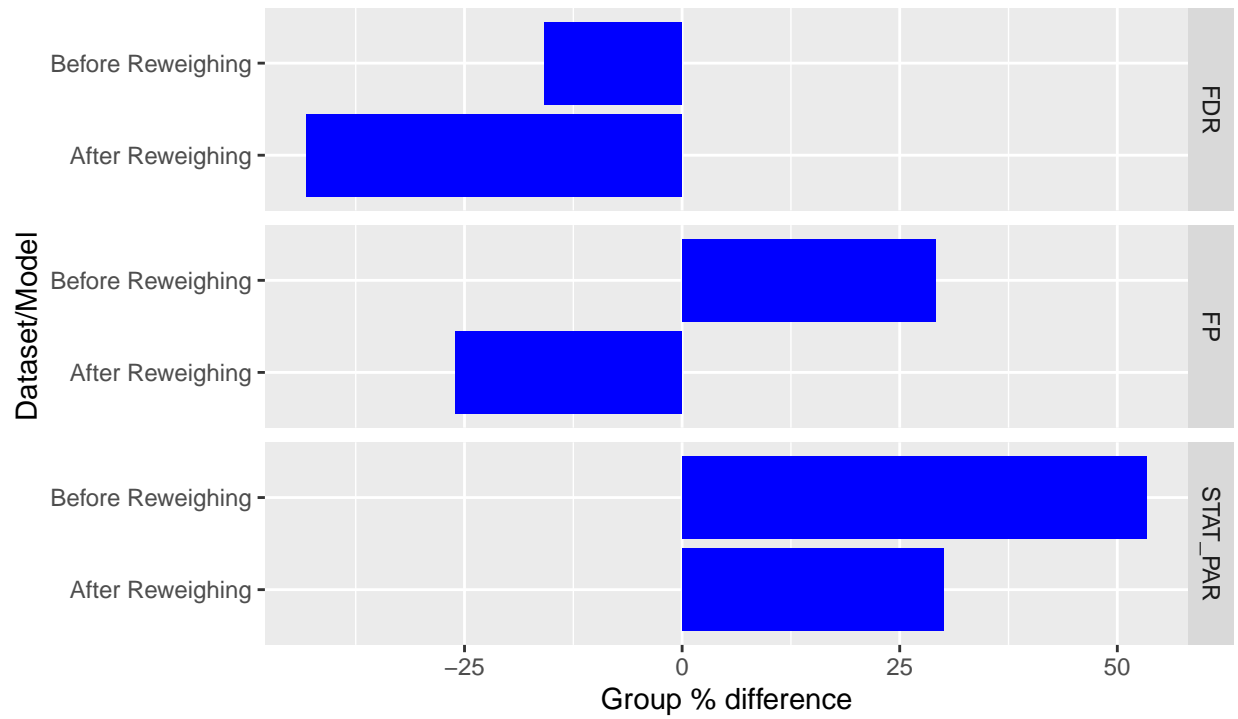
Age < 50: Development of Fairness Metrics from the Citys Perspective above 50 – below 50



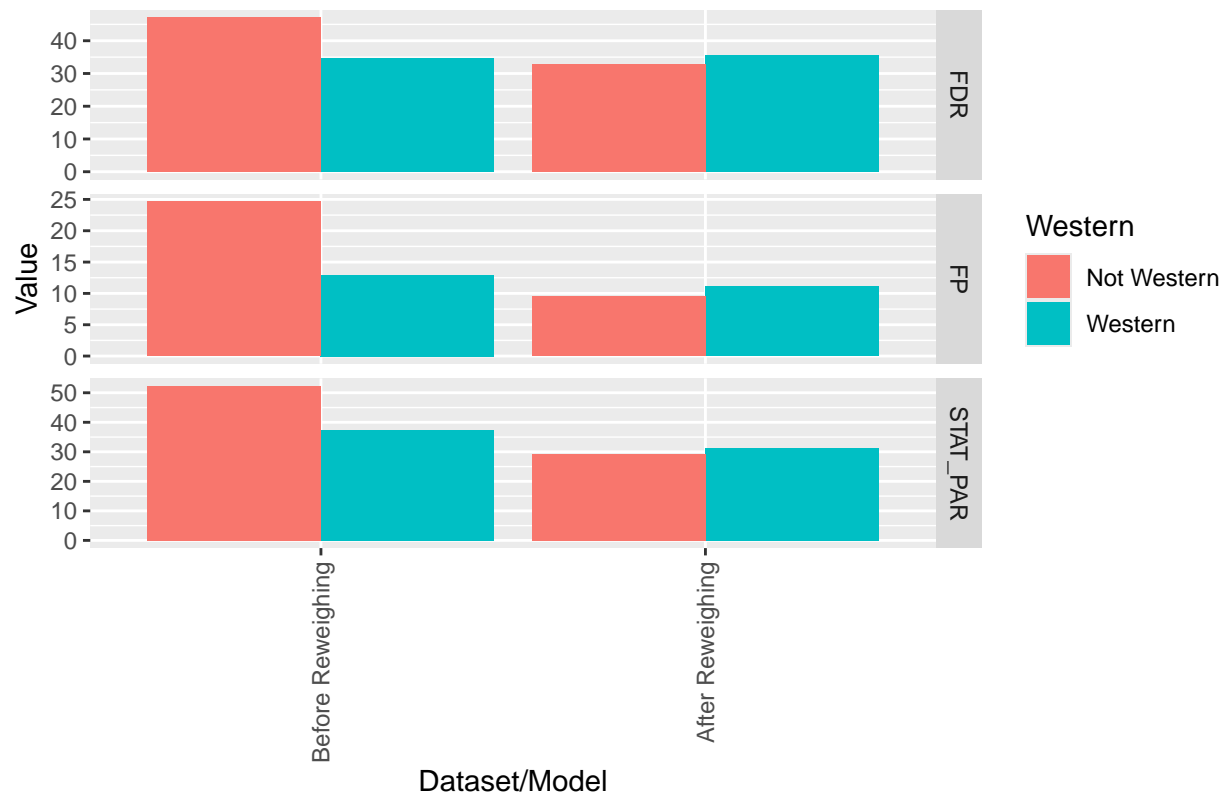
Dutch: Development of Fairness Metrics from the City's Perspective

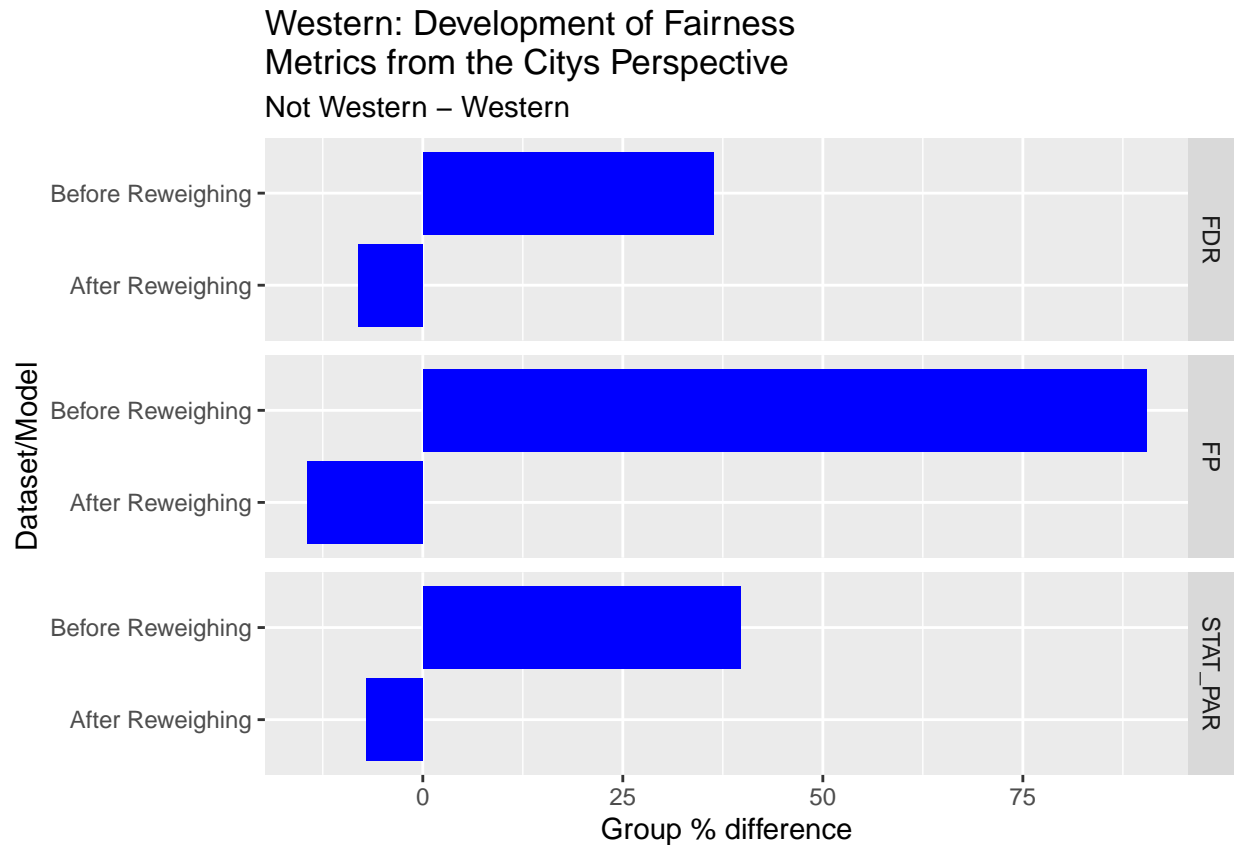


Dutch: Development of Fairness Metrics from the Citys Perspective Not Dutch – Dutch



Western: Development of Fairness Metrics from the Citys Perspective





RQ2 Comparing before and after reweighing

Amsterdam realized that its model exhibited bias, according to its bias definition (FP), when it deployed it in a pre-pilot (which really was a virtual pilot). The city decided to reweigh the training data to decrease the impact of observations on the model that drove the bias. Let's have a look at how the two models fared when tested against a more complete set of fairness metrics and datasets.

```
cms_model_comp <- cms_long %>%
  filter(Metric %in% c('PRED_P', 'FP', 'FDR', 'FPR'),
         Feature_EN %in% c('gender', 'Age < 30', 'Age < 50', 'Dutch', 'IsFulltimeParent')) %>%
  mutate(order = case_when(Model == 'BR' ~ 1,
                           Model == 'AR' ~ 2,
                           .default = NA),
         model_long = case_when(Model == 'BR' ~ 'Before reweighing',
                              Model == 'AR' ~ 'After reweighing',
                              .default = NA),)

for(characteristic in unique(cms_model_comp$Feature_EN)){
  cms_char <- cms_model_comp %>%
    filter(Feature_EN == characteristic)
  print(characteristic)

  feature_val_privileged <- unique(cms_char[cms_char$is_privileged_group == 1,]$Feature_Value_EN)[1]
  feature_val_unprivileged <- unique(cms_char[cms_char$is_privileged_group == 0,]$Feature_Value_EN)[1]
```

```

cms_char_diff <- cms_char %>%
  dplyr::select(-Feature_Value, -group_size, -is_privileged_group) %>%
  pivot_wider(names_from = 'Feature_Value_EN', values_from = 'Value') %>%
  #using Amsterdam's difference op here, though not sure the ref cat is always the same
  mutate(Diff = 100*(.data[[feature_val_unprivileged]] - .data[[feature_val_privileged]])/.data[[feature_val_privileged]])

p3 <- ggplot(cms_char_diff, aes(x = reorder(Model_EN, order), y = Diff, shape = Dataset))+
  geom_point(size = 3)+
  facet_wrap(~Metric, scales = 'free')+
  geom_hline(yintercept = 0, color = 'red', scales = 'free_y')+
  labs(x = 'Model', y = 'Group % Difference',
       title = paste0(characteristic, ': Performance Difference by Model Type'),
       subtitle = paste0(feature_val_unprivileged, ' - ', feature_val_privileged))

print(p3)

ggsave(paste0('../output/rq2_p3_', characteristic, '.png'), plot = p3, width = 10, height = 8)
}

```

```
## [1] "gender"
```

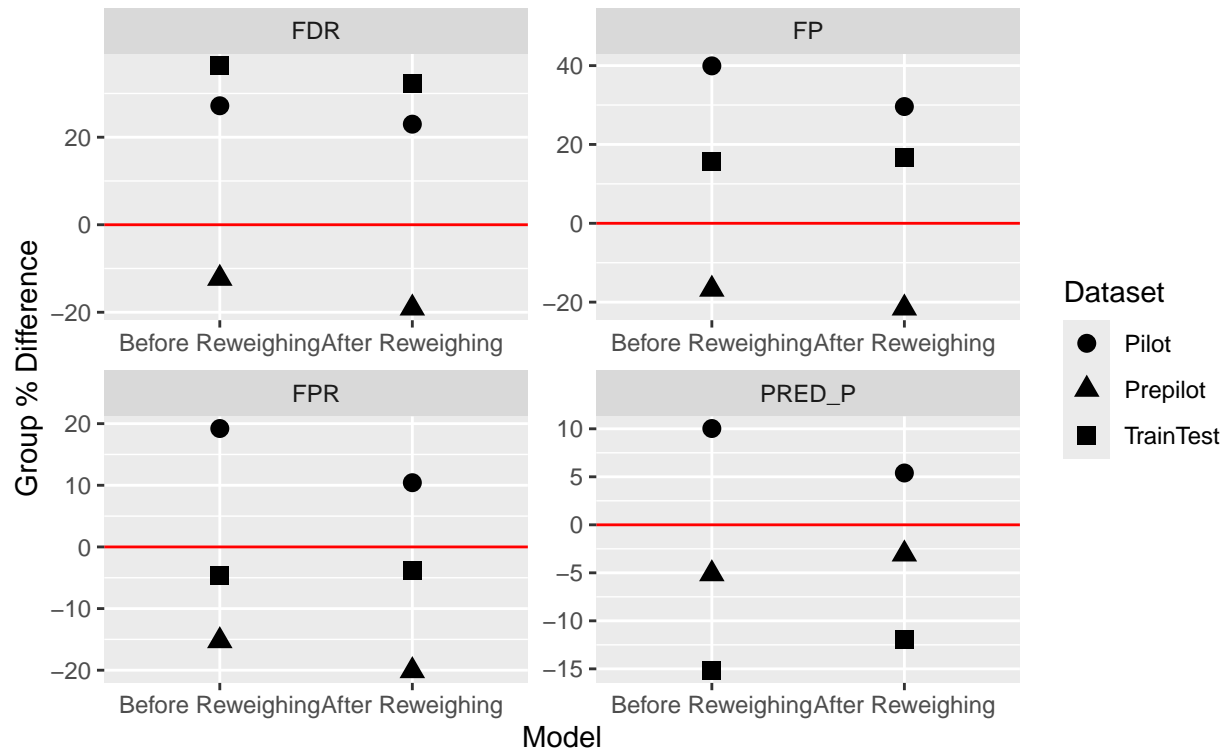
```
## Warning in geom_hline(yintercept = 0, color = "red", scales = "free_y"):  
## Ignoring unknown parameters: 'scales'
```

```
## [1] "Age < 30"
```

```
## Warning in geom_hline(yintercept = 0, color = "red", scales = "free_y"):  
## Ignoring unknown parameters: 'scales'
```

gender: Performance Difference by Model Type

F – M



```
## Warning: Removed 8 rows containing missing values or values outside the scale range
## ('geom_point()').
```

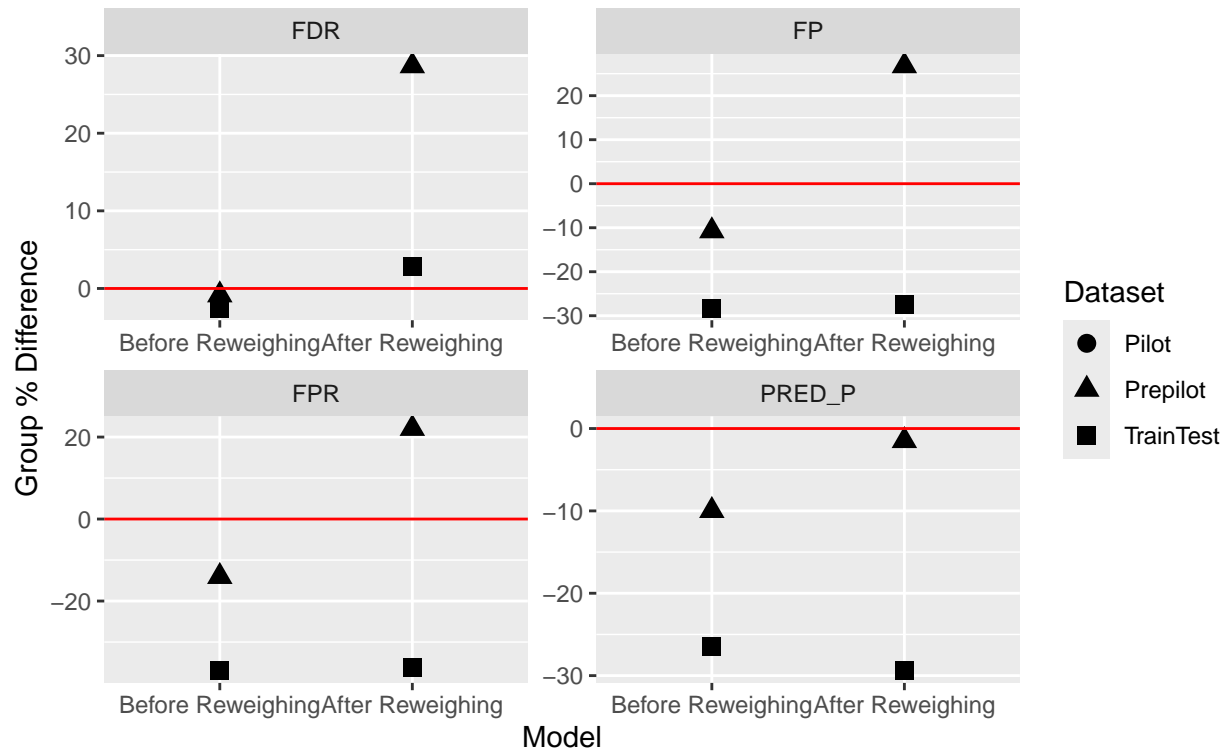
```
## Warning: Removed 8 rows containing missing values or values outside the scale range
## ('geom_point()').
```

```
## [1] "Age < 50"
```

```
## Warning in geom_hline(yintercept = 0, color = "red", scales = "free_y"):
## Ignoring unknown parameters: 'scales'
```

Age < 30: Performance Difference by Model Type

above 30 – below 30



```
## Warning: Removed 4 rows containing missing values or values outside the scale range
## ('geom_point()').
```

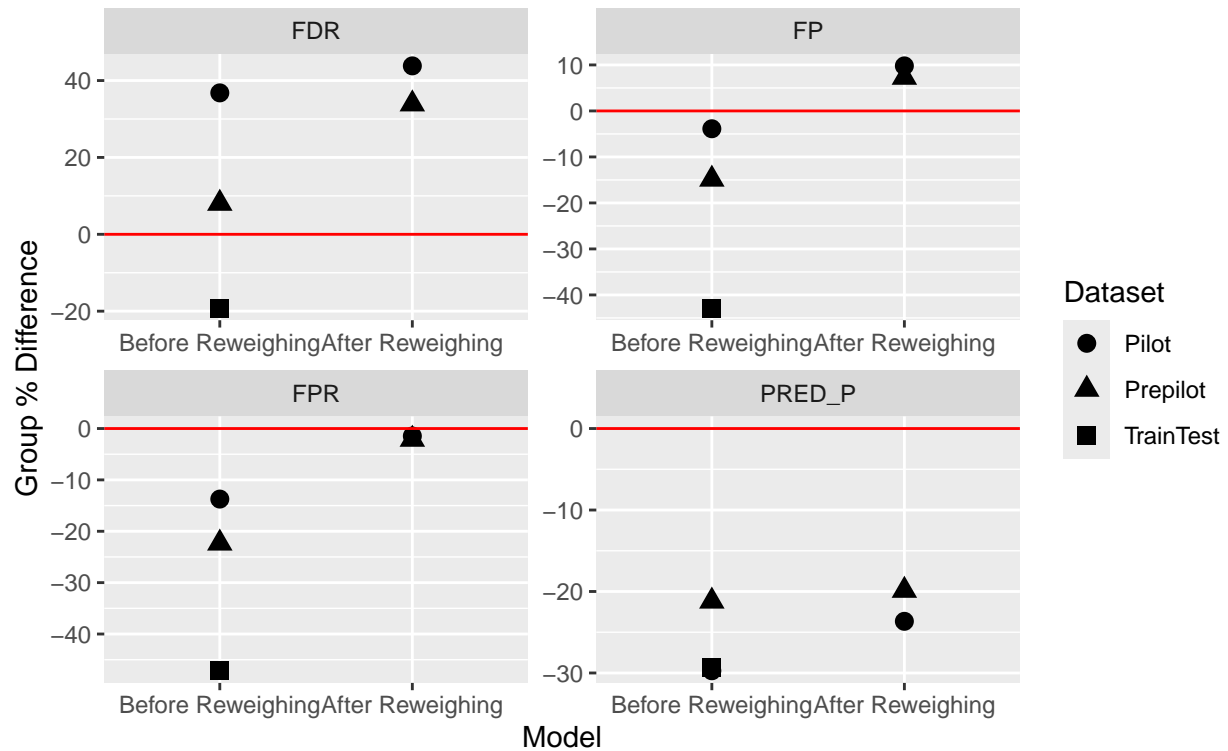
```
## Warning: Removed 4 rows containing missing values or values outside the scale range
## ('geom_point()').
```

```
## [1] "Dutch"
```

```
## Warning in geom_hline(yintercept = 0, color = "red", scales = "free_y"):
## Ignoring unknown parameters: 'scales'
```

Age < 50: Performance Difference by Model Type

above 50 – below 50



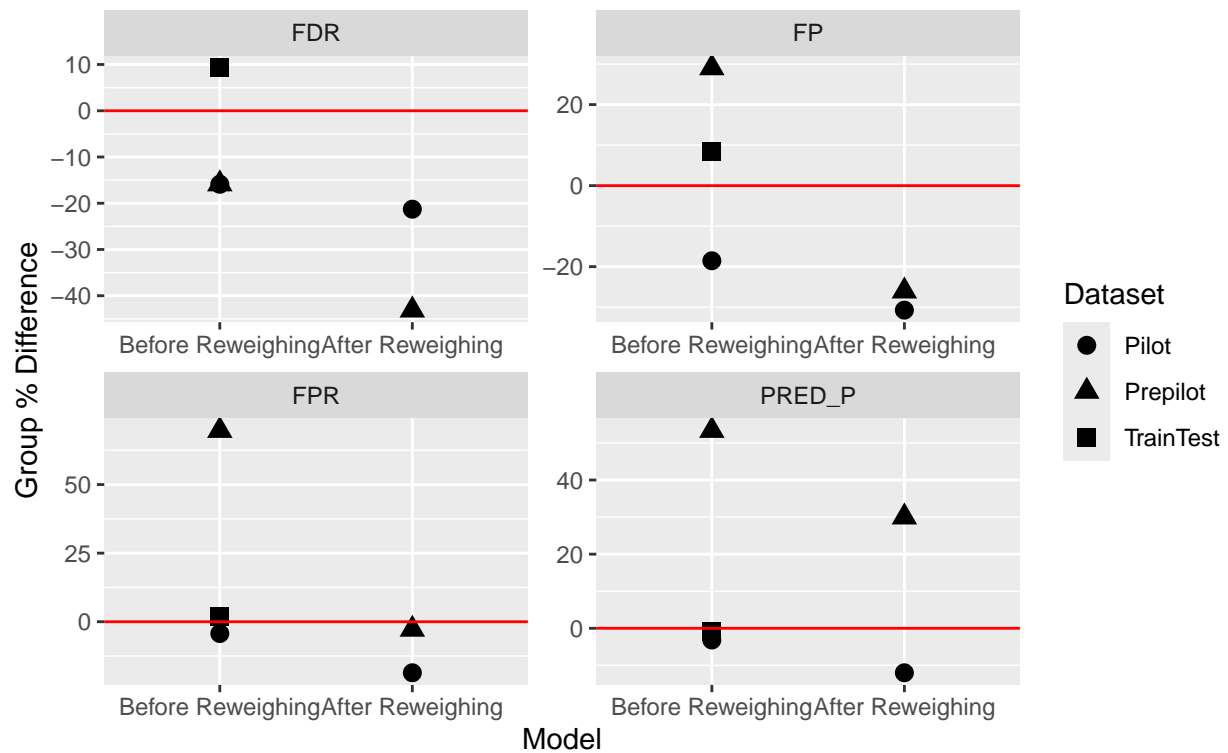
```
## Warning: Removed 4 rows containing missing values or values outside the scale range
## ('geom_point()').
## Removed 4 rows containing missing values or values outside the scale range
## ('geom_point()').

## [1] "IsFulltimeParent"

## Warning in geom_hline(yintercept = 0, color = "red", scales = "free_y"):
## Ignoring unknown parameters: 'scales'
```

Dutch: Performance Difference by Model Type

Not Dutch – Dutch

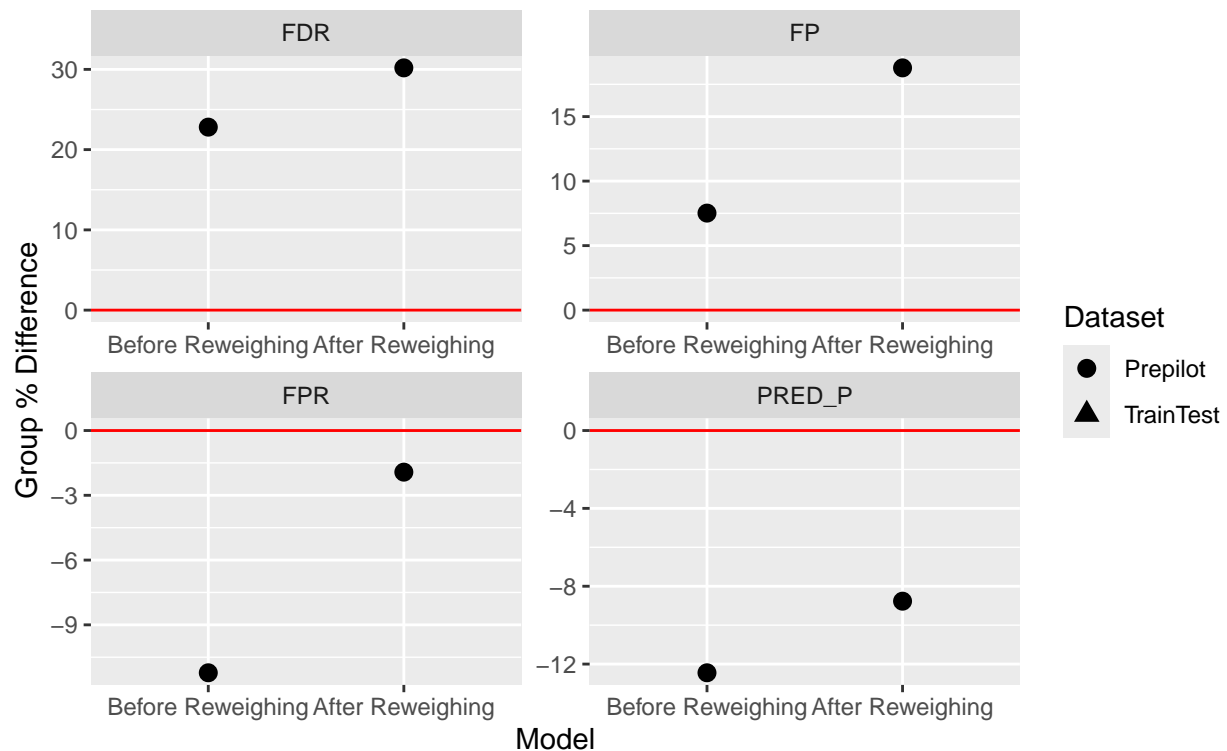


```
## Warning: Removed 8 rows containing missing values or values outside the scale range
## ('geom_point()').
```

```
## Warning: Removed 8 rows containing missing values or values outside the scale range
## ('geom_point()').
```

IsFulltimeParent: Performance Difference by Model Type

Full-time parent – Not full-time parent



RQ3 Error rate in Pilot

When looking at some of the graphs above, it stands out to me that the share of FPs jumped up substantially in the pilot. It seems useful to dig into that a bit more and find out if the rise in FP indicates a general deterioration of the model when confronted with pilot data and whether this potential deterioration is concentrated among particular groups.

```
cms_error <- cms_long %>%
  filter(stage %in% c('Prepilot/BR', 'Prepilot/AR', 'Pilot/BR', 'Pilot/AR'),
         Metric %in% c('FP', 'FN', 'ERROR'),
         Feature_EN %in% c('gender', 'Age < 30', 'Age < 50', 'Dutch', 'IsFulltimeParent')) %>%
  mutate(order = case_when(stage == 'Prepilot/BR' ~ 1,
                           stage == 'Prepilot/AR' ~ 2,
                           stage == 'Pilot/BR' ~ 3,
                           stage == 'Pilot/AR' ~ 4,
                           .default = NA))

for(characteristic in unique(cms_error$Feature_EN)){
  cms_char <- cms_error %>%
    filter(Feature_EN == characteristic)

  p4 <- ggplot(cms_char, aes(x = reorder(stage, order), y = Value, fill = Metric))+
    geom_bar(stat = 'identity', position = position_dodge())+
    facet_wrap(~Feature_Value_EN)+
    labs(x = 'Dataset/Model', y = '% Error',
```

```

    title = paste0(characteristic, ': Development of Error rates across model development'),
    fill = 'Error Metric')+
    theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
print(p4)

ggsave(paste0('../output/rq3_p4_error_', characteristic, '.png'), plot = p4, width = 10, height = 8)
}

```

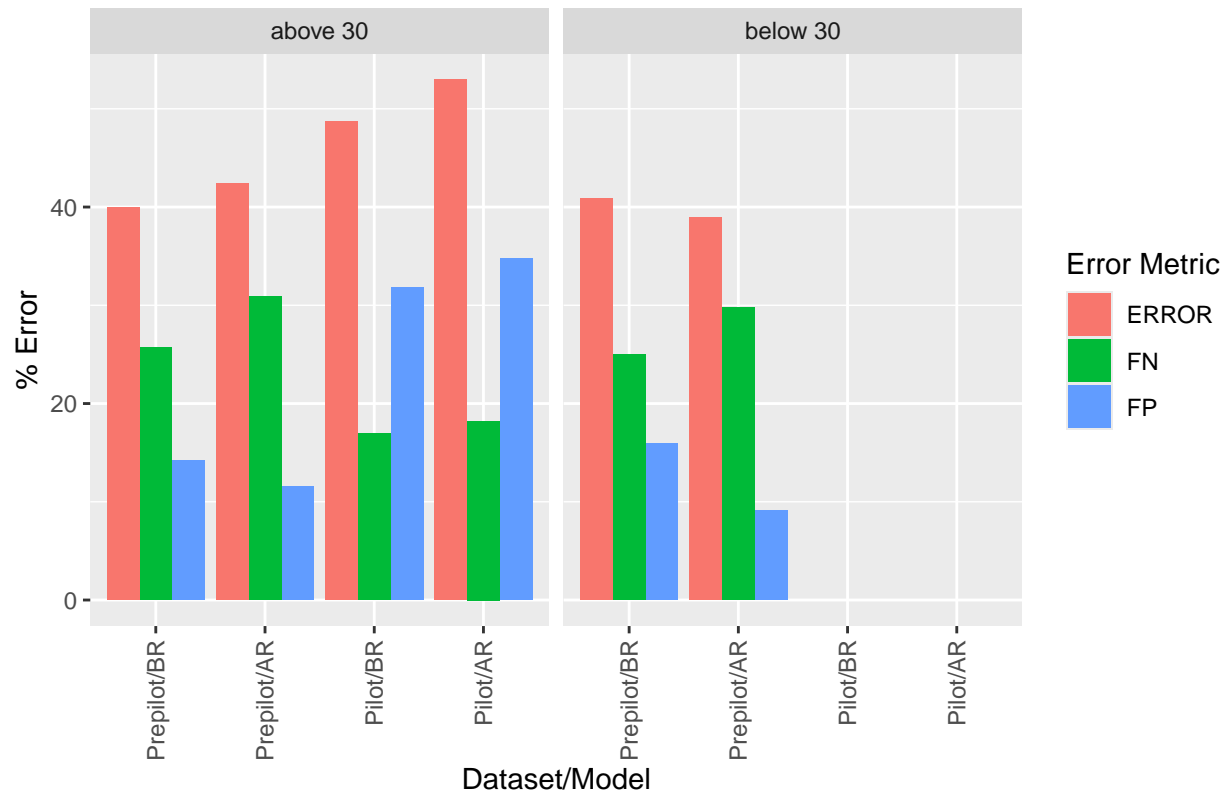


```

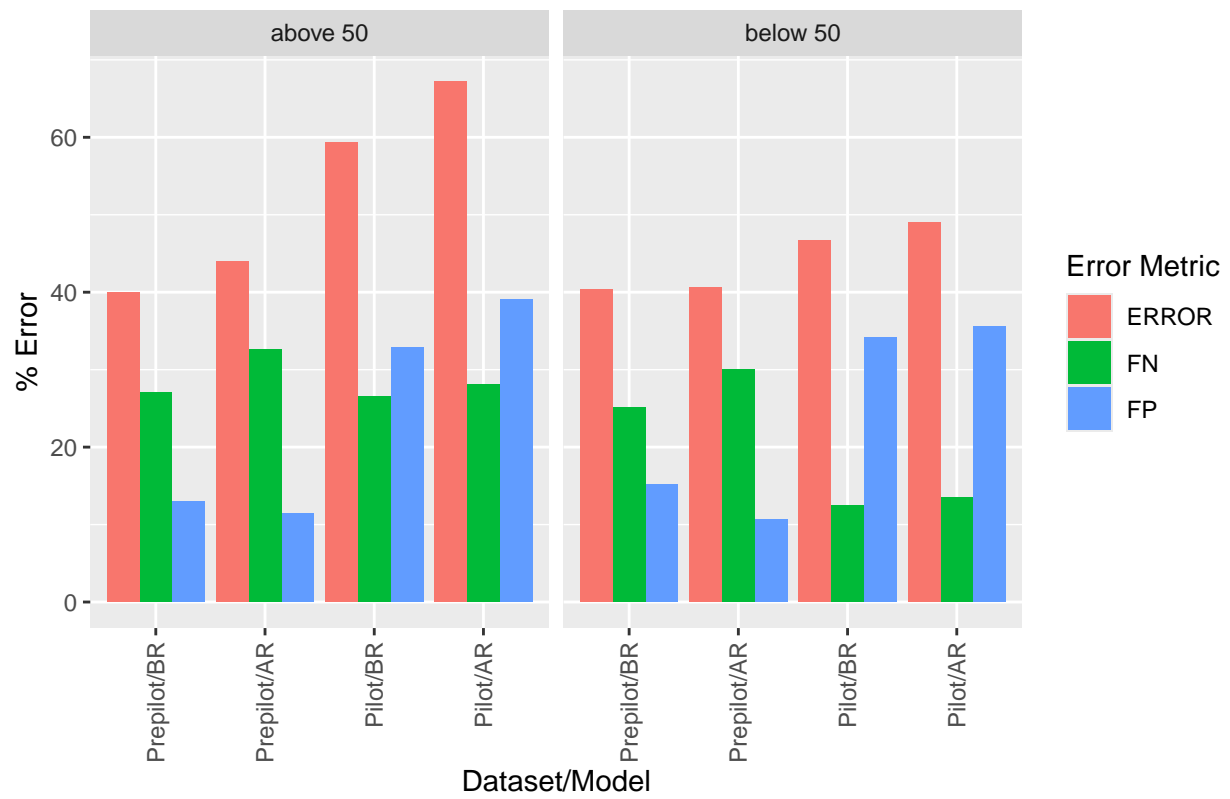
## Warning: Removed 6 rows containing missing values or values outside the scale range
## ('geom_bar()').
## Removed 6 rows containing missing values or values outside the scale range
## ('geom_bar()').

```

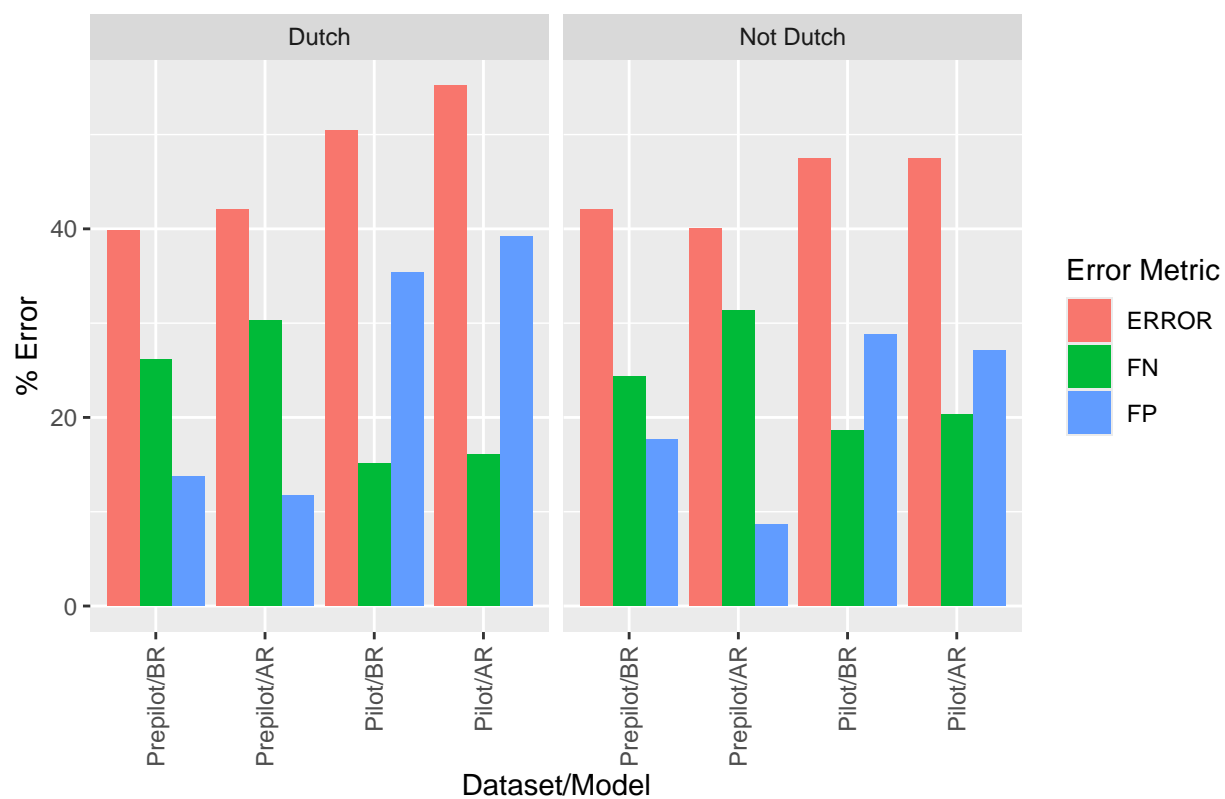

Age < 30: Development of Error rates across model development



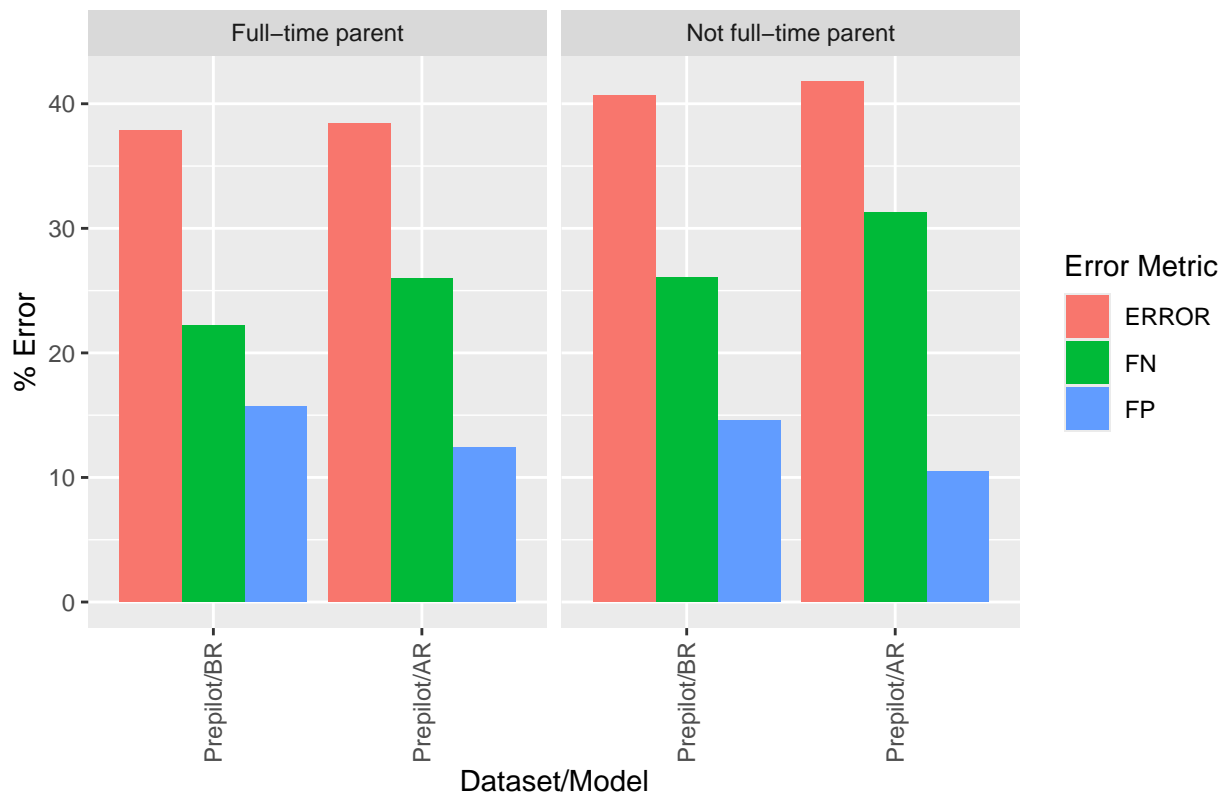
Age < 50: Development of Error rates across model development



Dutch: Development of Error rates across model development



IsFulltimeParent: Development of Error rates across model development



Feature importance

Along with the classification, the model provided caseworkers with the three most important features used by the model to come to its determination. Loek provided us access to the most important feature by demographic group. This allows us 1) to see if caseworkers could deduce beneficiary characteristics from the highlighted features, potentially activating their biases, and 2) whether the model used different features for different demographic groups in coming to its determination. The latter could be concerning under due process considerations.

```
feature_counts_restricted <- feature_counts %>%
  filter(Feature_EN %in% c("gender", "IsFulltimeParent", "Dutch")) %>%
  group_by(Feature_EN, Feature_Value_EN, dataset) %>%
  arrange(desc(share)) %>%
  mutate(rank = dense_rank(desc(share))) %>%
  slice_max(n = 5, order_by = share) %>%
  ungroup() %>%
  filter(!is.na(share))

for(characteristic in unique(feature_counts_restricted$Feature_EN)){
  feature_counts_char <- feature_counts_restricted %>%
    filter(Feature_EN == characteristic)

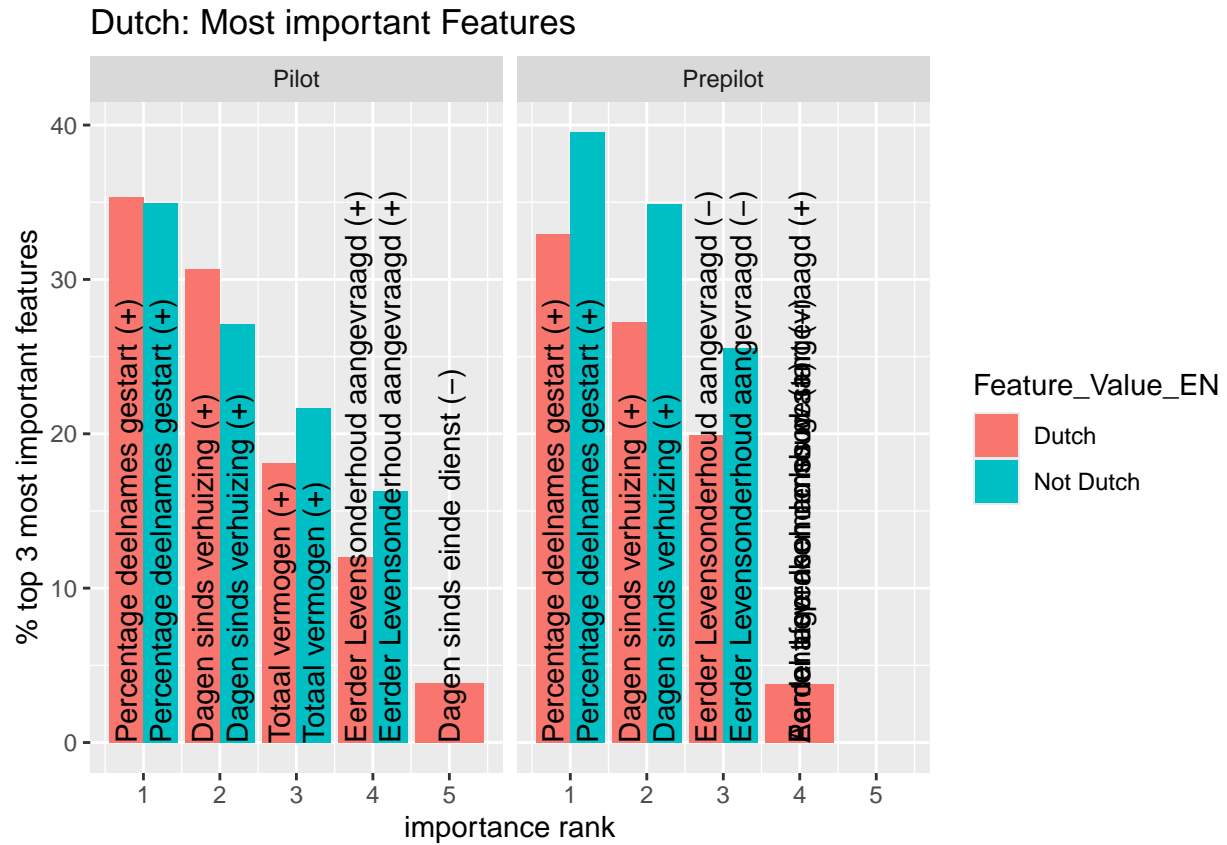
  p5 <- ggplot(feature_counts_char, aes(x = rank, y = share, fill = Feature_Value_EN, label = Important
    geom_bar(stat = 'identity', position = position_dodge()) +
    geom_text(aes(y = 0), hjust = 0, angle = 90, position = position_dodge(width = .9)) +
```

```

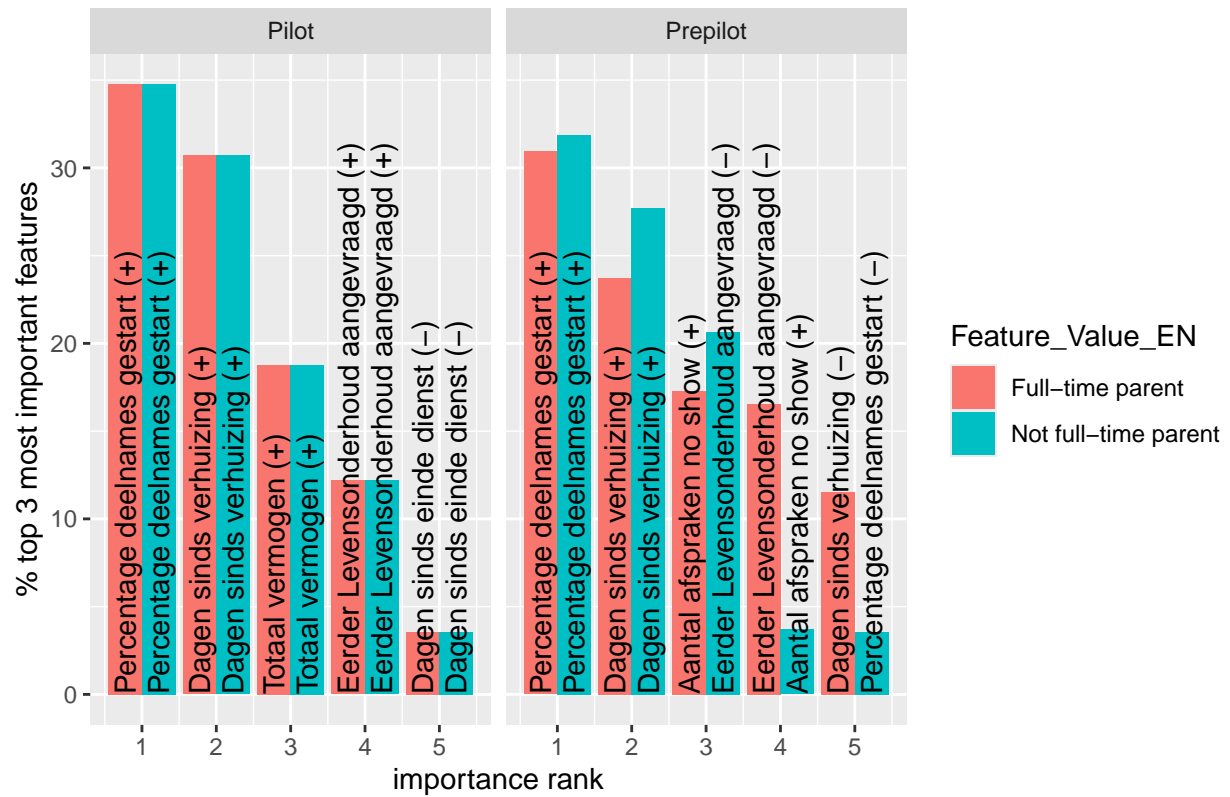
facet_grid(~dataset)+
labs(x = 'importance rank',
     y = '% top 3 most important features',
     title = paste0(characteristic, ': Most important Features'))

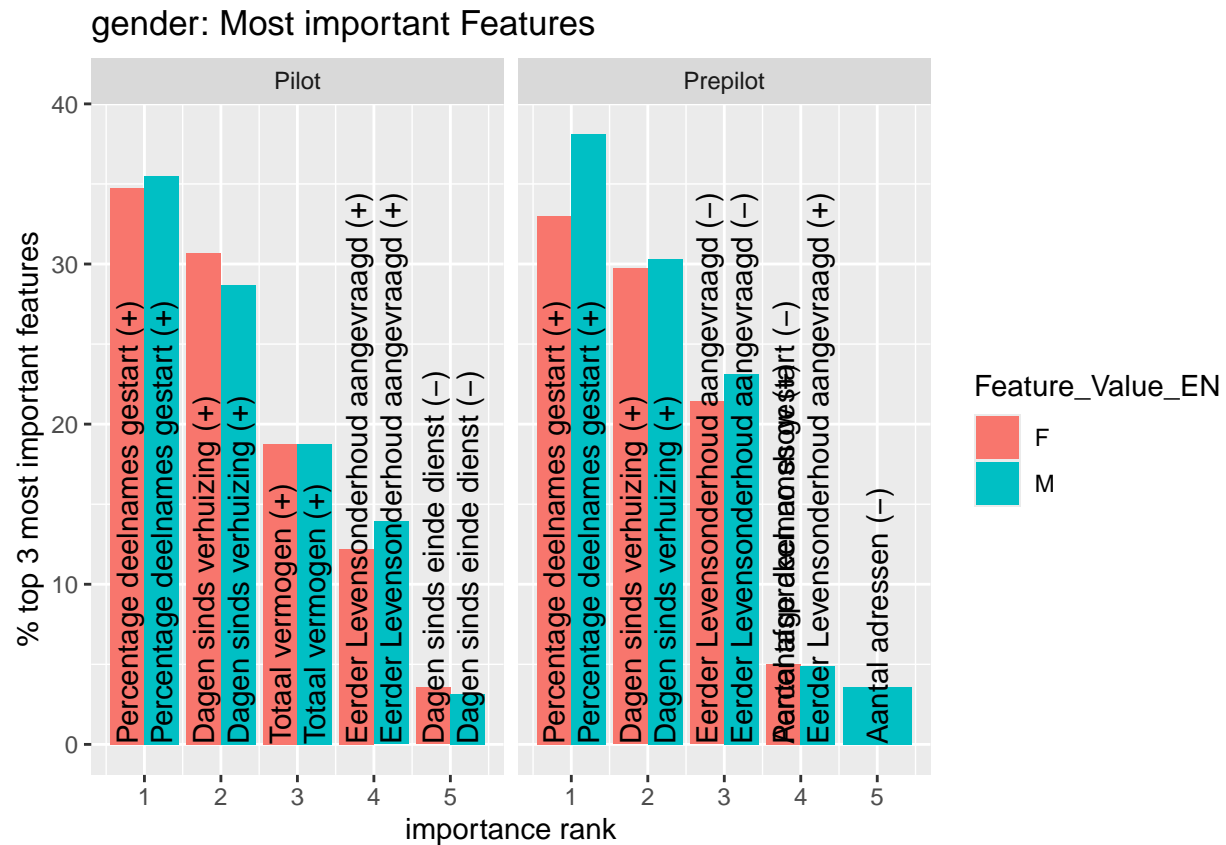
print(p5)
ggsave(paste0('../output/rq4_p5_feature_importance_', characteristic, '.png'), plot = p5, width = 10,
}

```



IsFulltimeParent: Most important Features





Impossibility Theorem

TODO: placeholder to build graphs that illustrate the impossibility theorem. Incidentally, the data above illustrates that it's really a tradeoff between predictive parity, One_minus_PPV, and FPR (something i have embarrassingly often misstated).

Viz

Just a codeblock to extract data for viz or make plots prettier

```
cms_gender <- cms_long %>%
  filter(Feature == 'geslacht', Metric %in% c('TP', 'FP', 'TN', 'FN'), Dataset == 'Prepilot', Model ==
```