

# Supplementary Digital Content - The Performance of Hypotension Prediction Index May Be Overestimated Due to Selection Bias

Johannes Enevoldsen and Simon Tilma Vistisen

## Contents

Setup	1
Simulation of the selection problem	1
The ROC curve shape in the simulations differs slightly from those presented in the model-develop paper.	4
Create figure 3	6

## Setup

```
library(pROC)
library(tidyverse)
# install.packages("faux") # used to simulate data

# A few extra packages are used to create the simulation figure. We load these in
# the relevant section.

theme_set(theme_minimal(base_size = 12))
```

## Simulation of the selection problem

**OBS:** This simulation does not attempt to create realistic data or to accurately preset how Hypotension Prediction Index may have been developed.

We simulate 1000 paired observations, to show what the data selection described in the model-development paper does to the performance of a model that uses MAP as a predictor.

The specific parameters of the simulation are simply chosen to place most data in a relevant interval: between 55 and 95 mmHg. The correlation coefficient ( $r = 0.5$ ) is chosen to give MAP a

modest predictive power, similar to what is presented by Davies *et al* (<http://doi.org/10.1213/ANE.0000000000004121>) However, any reasonable distribution of data, and any correlation coefficient between 0 and ~0.9, will convey the intended message.

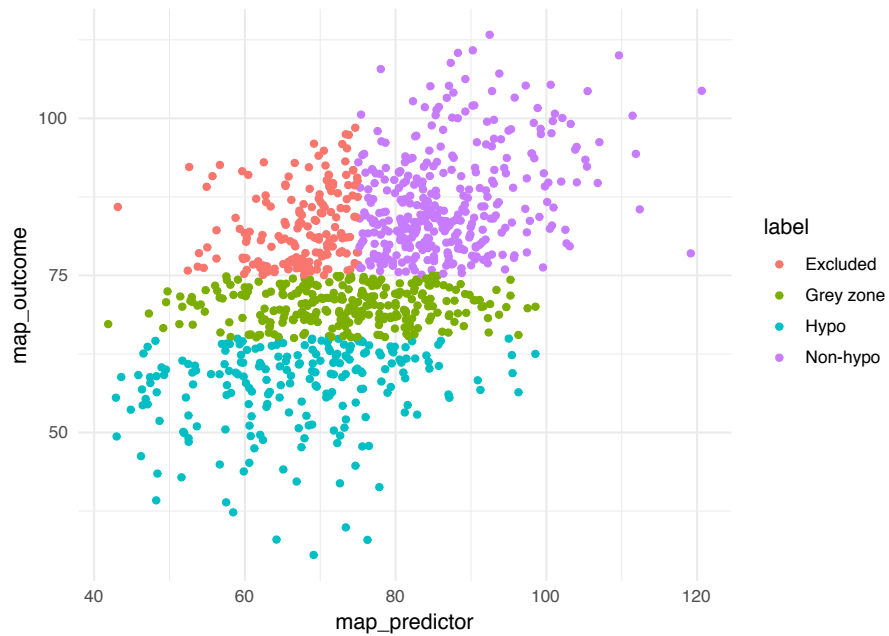
```
set.seed(12345)
correlated_map <- faux::rnorm_multi(n = 1000,
                                   mu = c(75, 75),
                                   sd = c(13, 13),
                                   r = 0.5,
                                   # "map_predictor" represents the current map value
                                   # "map_outcome" represents what map is 5 minutes from now.
                                   varnames = list("map_predictor", "map_outcome")
                                   )
```

We label the relevant zones of the data. Each row is the combination of an event (hypotensive or non-hypotensive) and the sample used to predict the event (map\_predictor).

```
d <- mutate(correlated_map,
            # If map_outcome is < than 65, this is represented as a hypotensive event
            hypo = map_outcome < 65,
            # If map_outcome is between 65 and 75, this is a greyzone event.
            grey = map_outcome >= 65 & map_outcome < 75,
            # If map_outcome and map_predictor is >= 75, this is a nonevent.
            nonhypo = map_outcome >= 75 & map_predictor >= 75
            )
```

Here is a plot to show the applied labels.

```
ggplot(d, aes(map_predictor, map_outcome, color = case_when(hypo ~ "Hypo",
                                                            grey ~ "Grey zone",
                                                            nonhypo ~ "Non-hypo",
                                                            TRUE ~ "Excluded")))) +
  geom_point() +
  labs(color = "label")
```



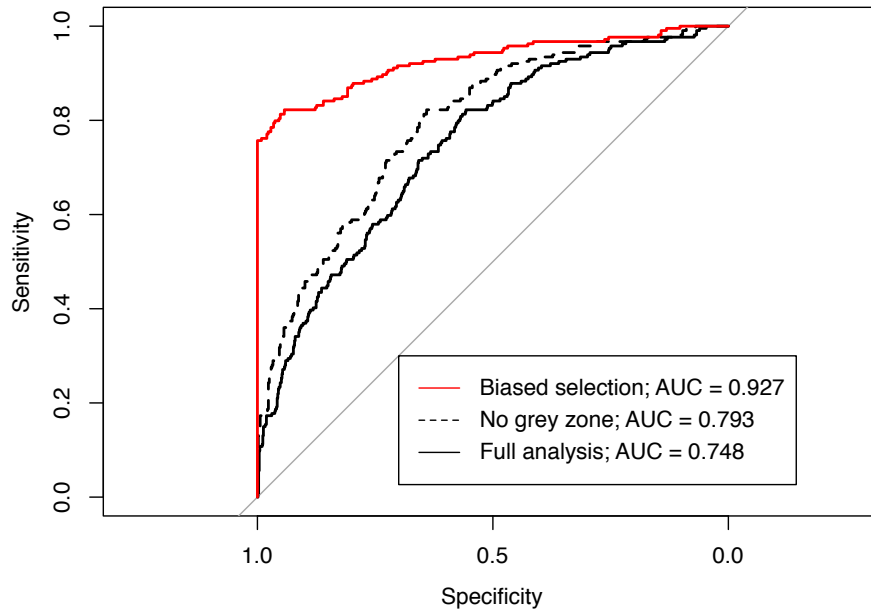
Now we can make the ROC analyses.

```
# With all data
roc_analysis_full <- roc(hypo~map_predictor, data = d)

# Excluding grey zone
roc_analysis_grey <- roc(hypo~map_predictor, data = filter(d, !grey))

# Include only data that is either event or non-event per model-development definition
roc_analysis_biased <- roc(hypo~map_predictor, data = filter(d, hypo | nonhypo))

plot(roc_analysis_full)
plot(roc_analysis_grey, lty = 2, add = TRUE)
plot(roc_analysis_biased, col = "red", add = TRUE)
legend(
  x = 0.7, y = 0.3,
  legend = c(
    sprintf("Biased selection; AUC = %.3f", auc(roc_analysis_biased)),
    sprintf("No grey zone; AUC = %.3f", auc(roc_analysis_grey)),
    sprintf("Full analysis; AUC = %.3f", auc(roc_analysis_full))
  ),
  lty = c(1, 2, 1),
  col = c("red", "black", "black")
)
```



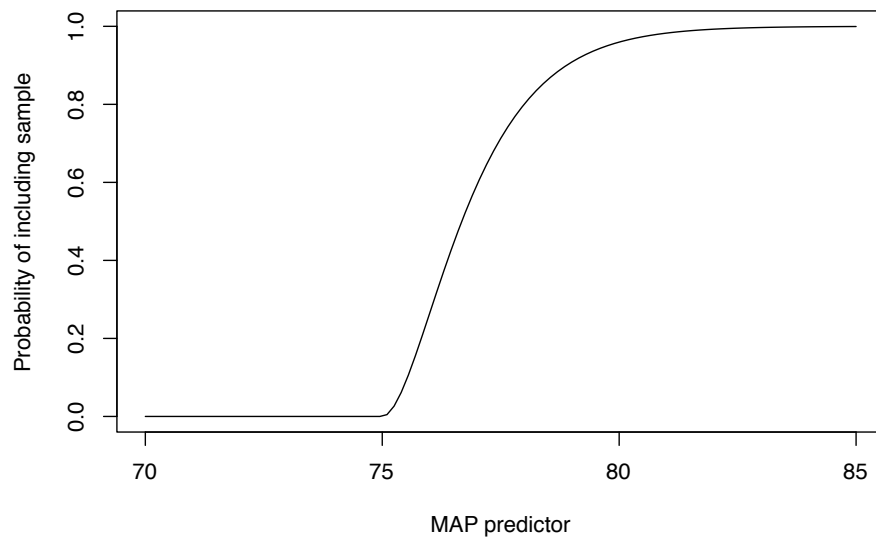
In the end of this document we paste the remaining code used to create the paper's figure 3.

## The ROC curve shape in the simulations differs slightly from those presented in the model-develop paper.

The ROC curve in the paper's fig. 2C shows a distinct decrease in specificity when the map threshold is  $> 75$  mmHg. In the model-development paper, the decrease in specificity is more gradual. (fig. 1). This difference can be explained by a combination of two effects. First, in the model development, all MAP values in a 30-minute period had to be  $> 75$  mmHg; due to the variation in MAP over 30 minutes, any specific measurement (here, the midpoint) is unlikely to be very close to 75 mmHg. Second, although HPI may be driven predominantly by MAP, other predictors influence HPI as well.

Here, we demonstrate the first explanation. We select non-hypotensive events based to the predictor MAP's distance from 75 mmHg. We use a gamma distribution to determine the probability that an observation should be included. (This is just one of many ways of making a simulation where non-hypotensive events with a predictor MAP close to 75 mmHg are unlikely).

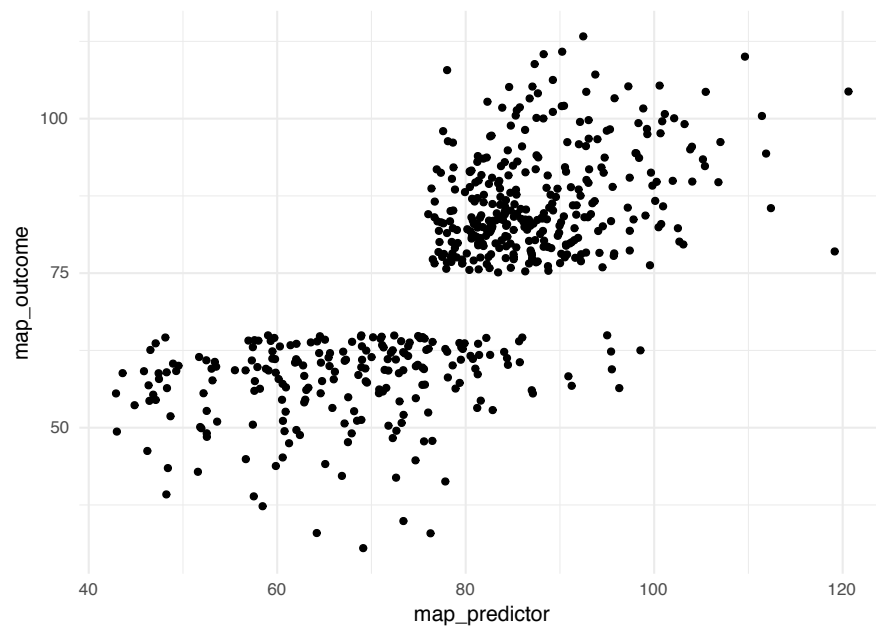
```
curve(pgamma(x-75, shape =2, rate = 1), 70, 85,
      xlab = "MAP predictor",
      ylab = "Probability of including sample")
```



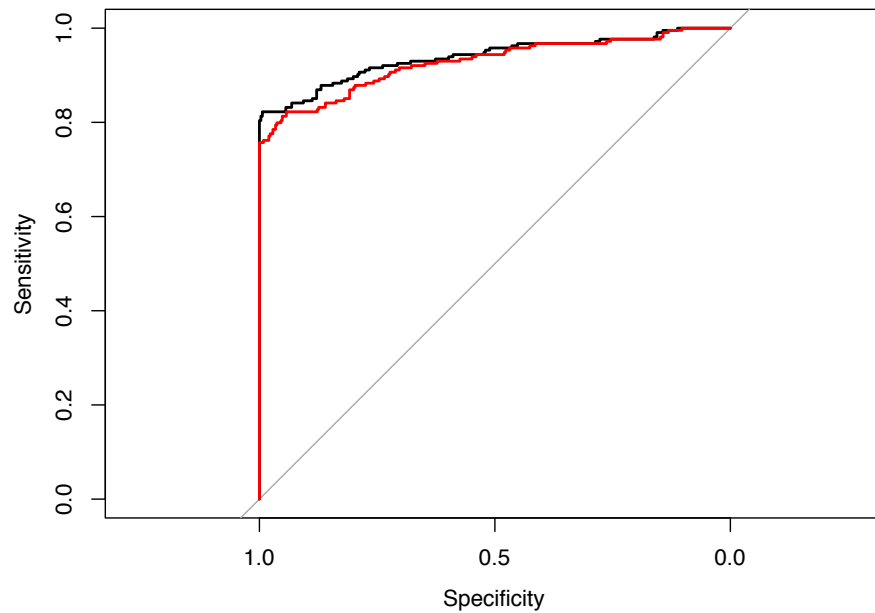
```
sampled <- rbernoulli(n=nrow(d), p = pgamma(d$map_predictor-75, shape = 2, rate = 1))

d2 <- filter(d, hypo | (nonhypo & sampled))

ggplot(d2, aes(map_predictor, map_outcome)) +
  geom_point()
```



```
roc_d2 <- roc(hypo ~ map_predictor, data = d2)
plot(roc_d2)
plot(roc_analysis_biased, add = TRUE, col = "red")
```



## Create figure 3

```
library(ggrepel)
library(patchwork)

# Set plot theme
# Plot theme
# Theme for plot
theme_set(
  theme_minimal(base_size = 11) +
    theme(
      plot.title = ggtext::element_textbox_simple(
        size = 10,
        vjust = 0,
        minheight = unit(3, "lines"), width = unit(2.5, "inches")
      ),
      axis.title = element_text(size = 9.5),
      axis.text = ggplot2::element_text(size = ggplot2::rel(0.8)),
      panel.grid.major = ggplot2::element_line(
        color = "#454545",
        size = 0.3,
        linetype = "dotted"
      ),
      panel.grid.minor = ggplot2::element_blank()
    )
)

plot_colors <- c("#2A6EBB", "#bb612a")
```

```

# First we extract all combinations of sensitivity, specificity and thresholds for
# all ROC analyses
roc_data_full <- as.data.frame(roc_analysis_full[c("sensitivities",
                                                  "specificities",
                                                  "thresholds")]) |>
  arrange(desc(specificities), sensitivities)

roc_data_grey <- as.data.frame(roc_analysis_grey[c("sensitivities",
                                                  "specificities",
                                                  "thresholds")]) |>
  arrange(desc(specificities), sensitivities)

roc_data_biased <- as.data.frame(roc_analysis_biased[c("sensitivities",
                                                      "specificities",
                                                      "thresholds")]) |>
  arrange(desc(specificities), sensitivities)

# We extract the rows with thresholds closest to 70, 75 and 80
# This is used for labeling the ROC curves.
roc_label_filter <- function(roc_data) {
  roc_data |>
    filter(
      row_number() == which.min(abs(thresholds - 70)) |
      row_number() == which.min(abs(thresholds - 75)) |
      row_number() == which.min(abs(thresholds - 80))
    )
}

roc_labels_full <- roc_label_filter(roc_data_full)
roc_labels_grey <- roc_label_filter(roc_data_grey)
roc_labels_biased <- roc_label_filter(roc_data_biased)

# Make the 3 ROC panels
roc_plot_full <- ggplot(roc_data_full, aes(1-specificities, sensitivities)) +
  geom_step() +
  geom_abline(slope = 1, intercept = 0, linetype = 2, color = "grey") +
  geom_label_repel(aes(label = glue::glue("MAP[now] < {round(thresholds, 0)}")),
    parse = TRUE,
    size = 3.1,
    box.padding = 0.1,
    point.size = 0.1,
    segment.color = "#555555",
    position = position_nudge_repel(y = -0.2, x=0.2),
    ylim = c(0.15, 1),
    direction = "y",
    label.size = NA,
    #vjust = 1,

```

```

      hjust = 0,
      family = "Helvetica",
      data = roc_labels_full
    ) +
    annotate("label",
      x = 1, y = 0,
      label = sprintf("AUC = %.2f", auc(roc_analysis_full)),
      hjust = 1, vjust = -0,
      label.size = NA,
      family = "Helvetica",
      size = 3.5) +
    labs(x = "1 - specificity", y = "Sensitivity") +
    coord_equal()

roc_plot_grey <- ggplot(roc_data_grey, aes(1-specificities, sensitivities)) +
  geom_step() +
  geom_abline(slope = 1, intercept = 0, linetype = 2, color = "grey") +
  geom_label_repel(aes(label = glue::glue("MAP[now] < {round(thresholds, 0)}")),
    parse = TRUE,
    size = 3.1,
    box.padding = 0.1,
    point.size = 0.1,
    segment.color = "#555555",
    direction = "y",
    position = position_nudge_repel(y = -0.2, x=0.2),
    ylim = c(0.15, 1),
    label.size = NA,
    #vjust = 1,
    hjust = 0,
    family = "Helvetica",
    data = roc_labels_grey
  ) +
  annotate("label",
    x = 1, y = 0,
    label = sprintf("AUC = %.2f", auc(roc_analysis_grey)),
    hjust = 1, vjust = -0,
    label.size = NA,
    family = "Helvetica",
    #label.size = NA,
    size = 3.5) +
  labs(x = "1 - specificity", y = "Sensitivity") +
  coord_equal()

roc_plot_biased <- ggplot(roc_data_biased, aes(1-specificities, sensitivities)) +
  geom_step() +
  geom_abline(slope = 1, intercept = 0, linetype = 2, color = "grey") +
  geom_label_repel(aes(label = glue::glue("MAP[now] < {round(thresholds, 0)}")),

```



```

        parse = TRUE,
        size = 3.1,
        box.padding = 0.1,
        point.size = 0.3,
        segment.color = "#555555",
        position = position_nudge_repel(y = -0.1, x=0.2),
        ylim = c(0.15, 1),
        direction = "y",
        label.size = NA,
        #vjust = 1,
        hjust = 0,
        family = "Helvetica",
        data = roc_labels_biased
    ) +
    annotate("label",
            x = 1, y = 0,
            label = sprintf("AUC = %.2f", auc(roc_analysis_biased)),
            hjust = 1, vjust = -0,
            label.size = NA,
            family = "Helvetica",
            size = 3.5) +
    labs(x = "1 - specificity", y = "Sensitivity") +
    coord_equal()

# Make the 3 scatter plots
color_scale <- scale_color_manual(values = plot_colors,
                                  labels = c("Non-hypotensive events", "Hypotensive events"))

fill_scale <- scale_fill_manual(values = plot_colors,
                                labels = c("Non-hypotensive events", "Hypotensive events"))

x_limits <- round(range(d$map_predictor) + c(-1, 1), 0)

x_scale <- scale_x_continuous(breaks = seq(45, 105, by = 10),
                             limits = x_limits,
                             expand = c(0,0),
                             name = "MAP<sub>now</sub> (predictor) [mmHg]")

scatter_plot_full <- ggplot(d, aes(map_predictor, map_outcome, color = hypo)) +
  geom_point(size = 0.7, show.legend = FALSE) +
  geom_hline(yintercept = 65, color = "#555555") +
  scale_y_continuous(breaks = seq(45, 105, by = 10), expand = expansion(add = c(1,5))) +
  x_scale +

  labs(y = "MAP 5 minutes later (outcome) [mmHg]", color = "") +
  ggtitle(sprintf("***A** Full analysis (n=%d)", nrow(d))) +
  color_scale +

```

```

coord_equal() +
theme(axis.title.x = ggtext::element_markdown())

labels <- tribble(
  ~label,          ~hypo, ~x, ~y,
  "Hypotensive events", TRUE, Inf, -Inf,
  "Non-hypotensive events", FALSE, -Inf, Inf
)

event_labels <- geom_label(aes(x, y, label = label),
  size = 3.5,
  #label.size = NA,
  fill = alpha("white", 0.5),
  vjust = "inward",
  hjust = "inward",
  family = "Helvetica",
  data = labels,
  show.legend = FALSE
)

scatter_plot_full_w_labs <- scatter_plot_full + event_labels

scatter_plot_grey <- scatter_plot_full %>% filter(d, !grey) +
  geom_hline(yintercept = 75, color = "#555555") +
  event_labels +
  ggtitle(sprintf("***B** Events where MAP (outcome) is between 65 and 75 mmHg (‘gray zone’) are excluded (n=%d)", n=

scatter_plot_biased <- scatter_plot_grey %>% filter(d, hypo | nonhypo) +
  annotate("segment",
    x = 75, xend = 75,
    y = 75, yend = 105,
    color = "#555555"
  ) +
  event_labels +
  ggtitle(sprintf("***C** Samples corresponding to non-hypotensive events must have a MAP (predictor) ≥ 75 mmHg (n=%d)", n=

# Density plots
get_density <- function(data, from = from) {
  dens <- density(data$map_predictor, adjust = 2, n = 100, from = from)
  data.frame(map_predictor = dens$x, density = dens$y)
}

get_grouped_density <- function(data, from_hypo = x_limits[1], from_nonhypo = x_limits[1]) {
  bind_rows(
    "TRUE" = get_density(data[data$hypo == TRUE,], from = from_hypo),
    "FALSE" = get_density(data[data$hypo == FALSE,], from = from_nonhypo),
    .id = "hypo"
  )
}

```

```

) |> mutate(hypo = as.logical(hypo))
}

density_full <- get_grouped_density(d) |>
  ggplot(aes(map_predictor, y = density, fill = hypo, group = hypo)) +
  geom_area(alpha = 0.7, position = "identity", show.legend = FALSE) +
  # geom_vline(xintercept = c(70, 75, 80), color = "#555555", linetype = 2) +
  color_scale +
  fill_scale +
  labs(y = "Density") +
  x_scale +
  theme(axis.title.x = ggtext::element_markdown())

density_grey <- density_full %>% get_grouped_density(filter(d, !grey))

density_biased <- density_full %>% get_grouped_density(filter(d, hypo | nonhypo),
  from_nonhypo = 75)

# Combine plots
simulation_figure <- scatter_plot_full_w_labs + scatter_plot_grey + scatter_plot_biased +
  density_full + density_grey + density_biased +
  roc_plot_full + roc_plot_grey + roc_plot_biased +
  plot_layout(ncol = 3, byrow = TRUE, heights = c(2.5, 1, NA))

ggsave("figure3_simulation.pdf", simulation_figure, device = cairo_pdf, width = 9, height = 7)
ggsave("figure3_simulation.png", simulation_figure, bg = "white", width = 9, height = 7)

```