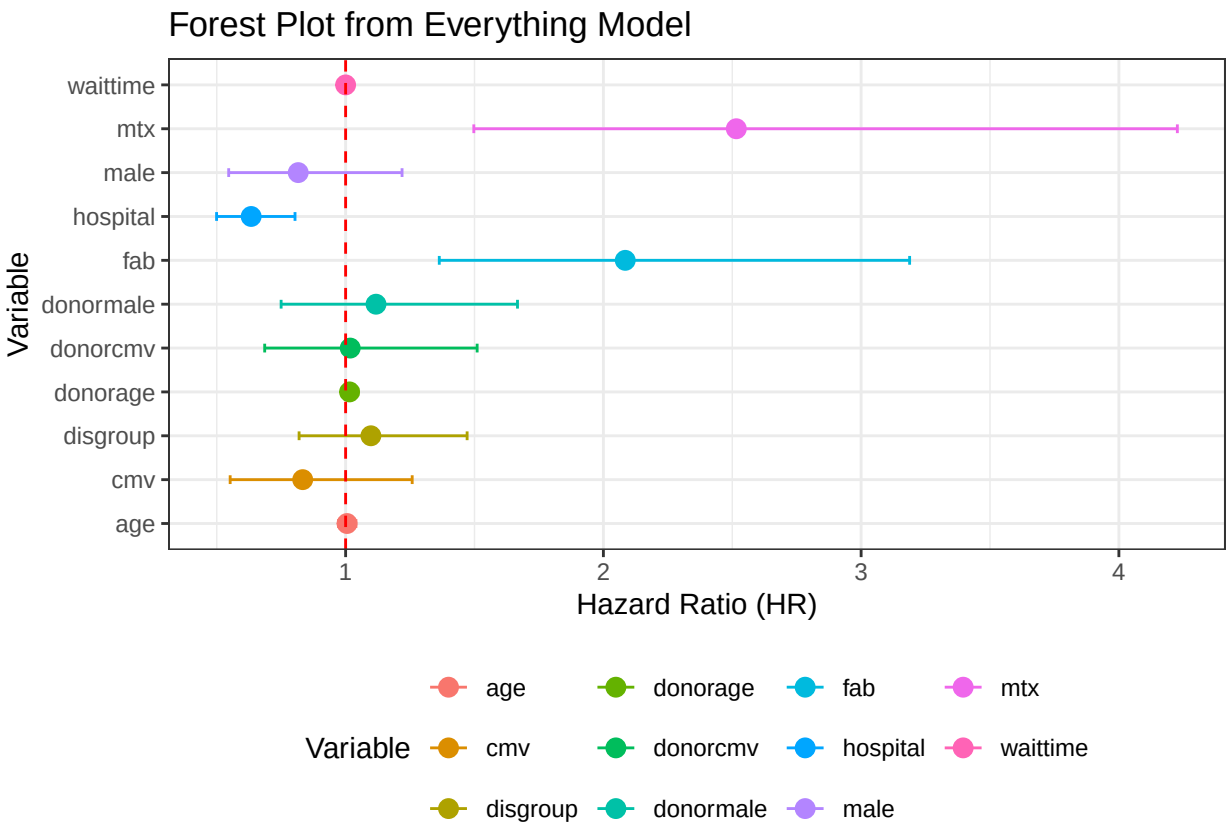
Directives 3 and 5

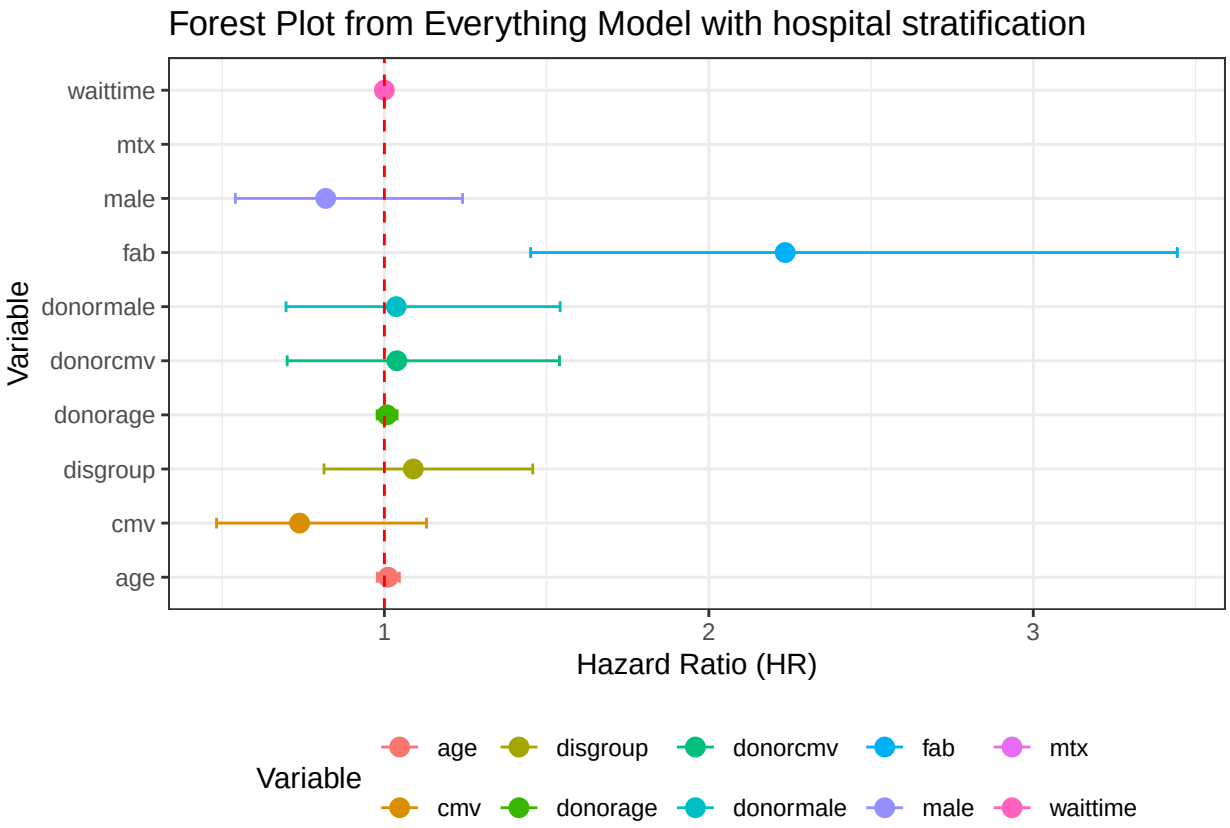
Department of Biostatistics @ University of Washington

Alejandro Hernandez

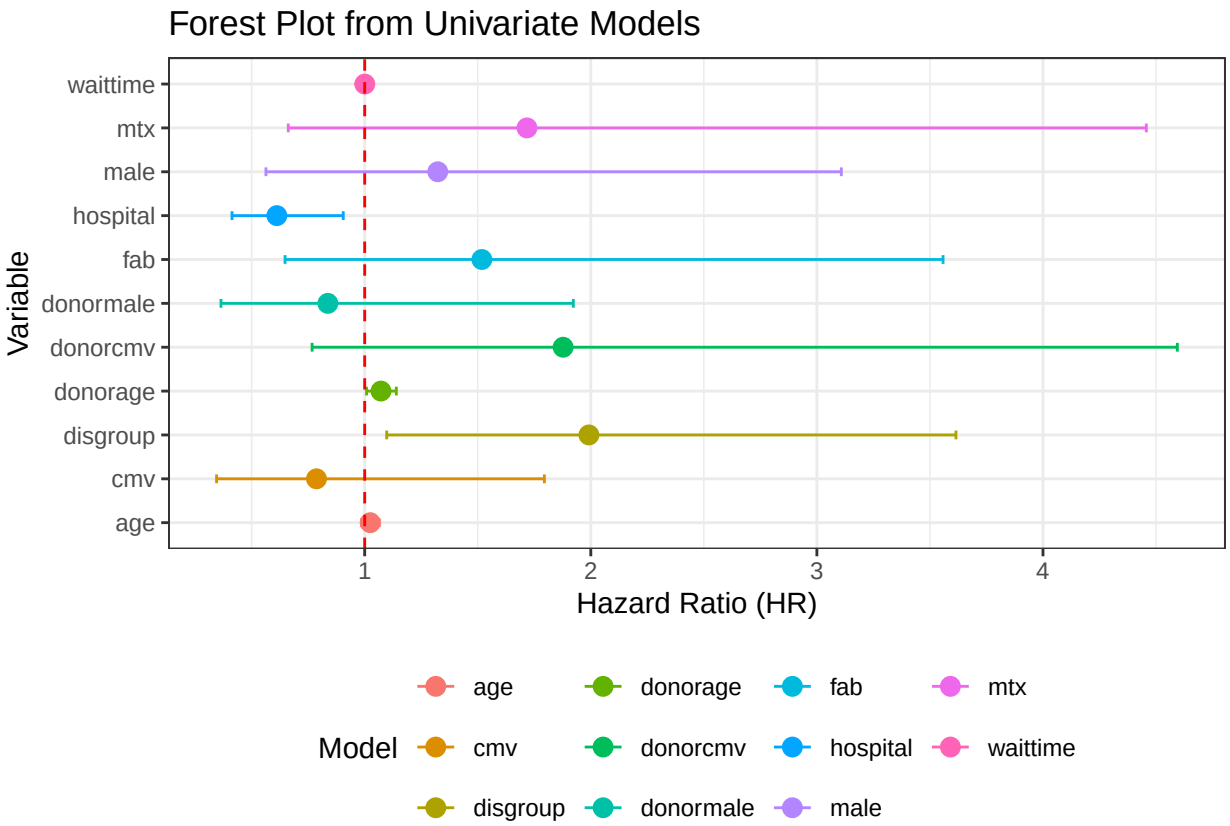
Winter Quarter 2025

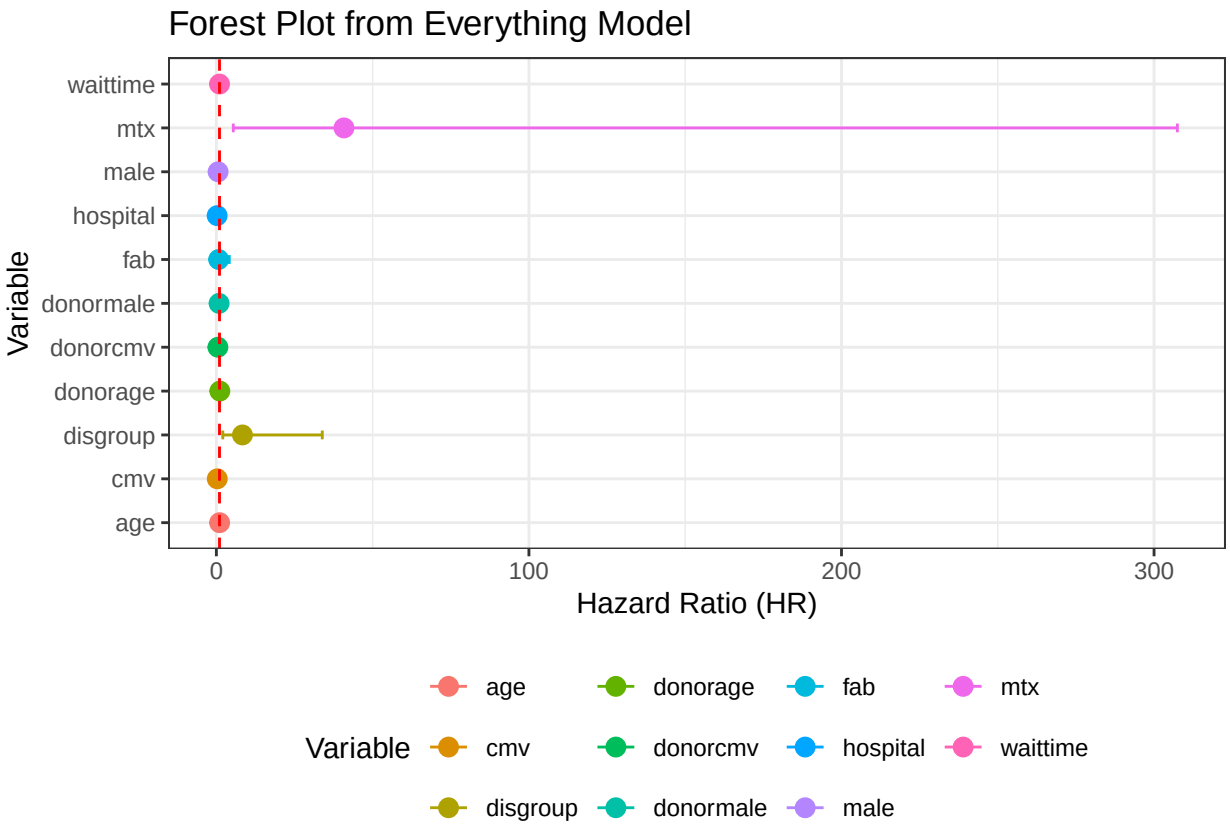
3. Are any of the measured baseline variables associated with differences in disease-free survival?

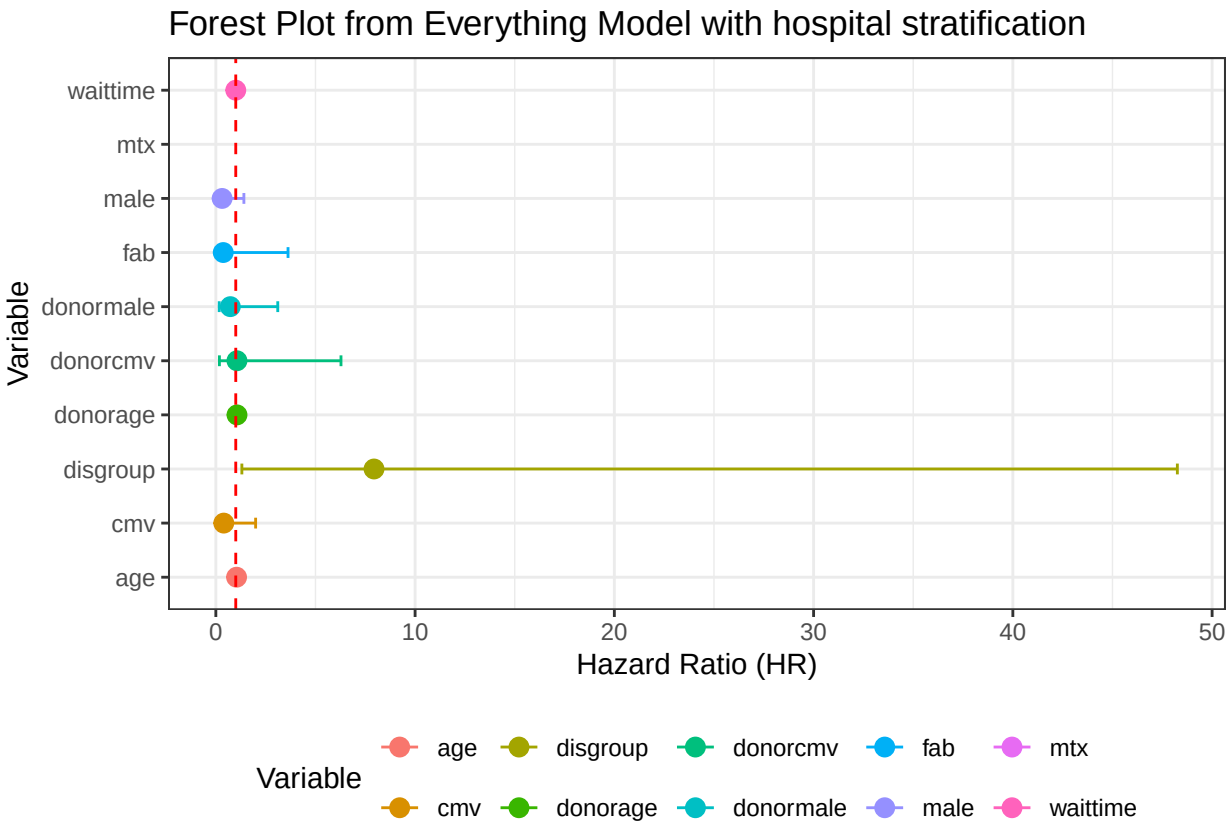




5. Among the patients who develop aGVHD, are any of the measured baseline factors associated with differences in disease-free survival?


Forest Plot from Univariate Models
waittime
mtx
male
hospital
fab
donormale
donorcmv
donorage
disgroup
cmv
age
Variable
1
2
3
4
Hazard Ratio (HR)
Model
age
cmv
disgroup
donorage
donorcmv
donormale
fab
hospital
male
mtx
waittime


Forest Plot from Everything Model
waittime
mtx
male
hospital
fab
donormale
donorcmv
donorage
disgroup
cmv
age
Variable
0
100
200
300
Hazard Ratio (HR)
Variable
age
cmv
disgroup
donorage
donorcmv
donormale
fab
hospital
male
mtx
waittime



End of report. Code appendix begins on the next page.

Code Appendix

```
# Clear environment
rm(list=ls())

# Setup options
knitr::opts_chunk$set(echo=FALSE, warning=FALSE, message=FALSE, results='hide')
options(knitr.kable.NA = '-', digits = 2)
labs = knitr::all_labels()
labs = labs[!labs %in% c("setup", "allcode")]
# Load relevant packages
library(survival)  # survival models
library(dplyr)     # data manipulation
library(broom)     # combine and reshape model output
library(ggplot2)   # data visualization

theme_set(theme_bw())

# Load data
bmt <- read.csv("../data/bmt_clean.csv")
dim(bmt) # 137 rows, 22 columns
names(bmt)

# Handle missing data (there is none)
anyNA(bmt)

# Create a measure of wait time in 3-month units, not days
bmt$waittime2 <- bmt$waittime / 90
# This is for a univariate log-rank test of a KM estimator
# Disease-free survival
dfs_surv <- with(bmt, survival::Surv(tdfs, deltadfs))
sort(dfs_surv)

## Log-rank tests of survival

# Fit KM models from baseline variables
model_list <- list(
  # Initial disease status
  disgroup = survdiff(dfs_surv ~ disgroup, bmt),
  fab = survdiff(dfs_surv ~ fab, bmt),
  waittime = survdiff(dfs_surv ~ waittime, bmt),
  mtx = survdiff(dfs_surv ~ mtx, bmt),
  hospital = survdiff(dfs_surv ~ hospital, bmt),
  # Patient
  age = survdiff(dfs_surv ~ age, bmt),
  male = survdiff(dfs_surv ~ male, bmt),
  cmv = survdiff(dfs_surv ~ cmv, bmt),
  # Donor
  donorage = survdiff(dfs_surv ~ donorage, bmt),
  donormale = survdiff(dfs_surv ~ donormale, bmt),
  donorcmv = survdiff(dfs_surv ~ donorcmv, bmt)
)
```

```r
# Extract results from each model
logrank_results <- lapply(model_list, function(model)
  {data.frame(logrank.pval = model$pvalue)}) %>%
  bind_rows(.id = "Variable") %>%
  arrange(logrank.pval)

logrank_results %>%
  mutate(logrank.pval = ifelse(logrank.pval < 0.001, "<0.001", round(logrank.pval,4)))
## Univariate proportional hazards models

# Fit PH models from baseline variables
model_list <- list(
  # Initial disease status
  disgroup = coxph(dfs_surv ~ disgroup, bmt),
  fab = coxph(dfs_surv ~ fab, bmt),
  waittime = coxph(dfs_surv ~ waittime, bmt),
  mtx = coxph(dfs_surv ~ mtx, bmt),
  hospital = coxph(dfs_surv ~ hospital, bmt),
  # Patient
  age = coxph(dfs_surv ~ age, bmt),
  male = coxph(dfs_surv ~ male, bmt),
  cmv = coxph(dfs_surv ~ cmv, bmt),
  # Donor
  donorage = coxph(dfs_surv ~ donorage, bmt),
  donormale = coxph(dfs_surv ~ donormale, bmt),
  donorcmv = coxph(dfs_surv ~ donorcmv, bmt)
)

# Extract results from each model
results <- lapply(model_list, function(model) {
  tidymodel <- broom::tidy(model, conf.int = TRUE, conf.level = 0.9)

  data.frame(
    HR = exp(tidymodel$estimate),
    HR.low = exp(tidymodel$conf.low),
    HR.high = exp(tidymodel$conf.high),
    logrank.pval = broom::glance(model)$p.value.sc[[1]], # Log-rank test
    LRT.pval = tidymodel$p.value  # Likelihood ratio test (LRT)
    )
  }
)
cox_results <- bind_rows(results, .id = "Model")

# Arrange variables by LRT p-value
cox_results %>% arrange(desc(LRT.pval))

# Variables with LRT p-values above or equal to 10%
cox_results %>% arrange(LRT.pval) %>% filter(LRT.pval <= 0.1)

# Forest plot of HR estimates
gg_univar <- ggplot(cox_results, aes(x=HR, y=Model, color=Model)) +
  geom_point(size = 3) + # Plot hazard ratio points
  geom_errorbarh(aes(xmin=HR.low, xmax=HR.high), height=0.2) +  # CI bars
```

```
  geom_vline(xintercept=1, linetype="dashed", color="red") +  # Reference line
  labs(title = "Forest Plot from Univariate Models",
       x = "Hazard Ratio (HR)",
       y = "Variable") +
  theme(legend.position = "bottom")

## Everything PH model

# Fit a single PH model from baseline variables
results <- coxph(dfs_surv ~
        # Initial disease status
        disgroup + fab + waittime + mtx + hospital +
        # Patient
        age + male + cmv +
        # Donor
        donorage + donormale + donorcmv,
      bmt) %>%
  tidy(conf.int = TRUE, conf.level = 0.9)

# Extract results
cox_results <- results %>%
  mutate(Variable = term,
         HR = exp(estimate),
         HR.low = exp(conf.low),
         HR.high = exp(conf.high),
         LRT.pval = p.value,
         .keep="none")

# Forest plot of HR estimates
ggplot(cox_results, aes(x=HR, y=Variable, color=Variable)) +
  geom_point(size = 3) + # Plot hazard ratio points
  geom_errorbarh(aes(xmin=HR.low, xmax=HR.high), height=0.2) +  # CI bars
  geom_vline(xintercept=1, linetype="dashed", color="red") +  # Reference line
  labs(title = "Forest Plot from Everything Model",
       x = "Hazard Ratio (HR)",
       y = "Variable") +
  theme(legend.position = "bottom")

## Everything model stratified by hospital

# Fit a single PH model from baseline variables
results <- coxph(dfs_surv ~ disgroup + fab + waittime + mtx + strata(hospital) +
                   age + male + cmv + donorage + donormale + donorcmv,
                 bmt) %>%
  tidy(conf.int = TRUE, conf.level = 0.9)

# Extract results
cox_results <- results %>%
  mutate(Variable = term, HR = exp(estimate), HR.low = exp(conf.low),
         HR.high = exp(conf.high), LRT.pval = p.value, .keep="none")

# Forest plot of HR estimates
ggplot(cox_results, aes(x=HR, y=Variable, color=Variable)) +
```

```
  geom_point(size = 3) + # Plot hazard ratio points
  geom_errorbarh(aes(xmin=HR.low, xmax=HR.high), height=0.2) +  # CI bars
  geom_vline(xintercept=1, linetype="dashed", color="red") +  # Reference line
  labs(title = "Forest Plot from Everything Model with hospital stratification",
       x = "Hazard Ratio (HR)",
       y = "Variable") +
  theme(legend.position = "bottom")

# Subset aGVHD patients
bmt_agvhd <- filter(bmt, deltaa==1)

# Disease-free survival
dfs_surv <- with(bmt_agvhd, survival::Surv(tdfs, deltadfs))
sort(dfs_surv)

## Log-rank tests of survival

# Fit KM models from baseline variables
model_list <- list(
  # Initial disease status
  disgroup = survdiff(dfs_surv ~ disgroup, bmt_agvhd),
  fab = survdiff(dfs_surv ~ fab, bmt_agvhd),
  waittime = survdiff(dfs_surv ~ waittime, bmt_agvhd),
  mtx = survdiff(dfs_surv ~ mtx, bmt_agvhd),
  hospital = survdiff(dfs_surv ~ hospital, bmt_agvhd),
  # Patient
  age = survdiff(dfs_surv ~ age, bmt_agvhd),
  male = survdiff(dfs_surv ~ male, bmt_agvhd),
  cmv = survdiff(dfs_surv ~ cmv, bmt_agvhd),
  # Donor
  donorage = survdiff(dfs_surv ~ donorage, bmt_agvhd),
  donormale = survdiff(dfs_surv ~ donormale, bmt_agvhd),
  donorcmv = survdiff(dfs_surv ~ donorcmv, bmt_agvhd)
)

# Extract results from each model
logrank_results <- lapply(model_list, function(model)
  {data.frame(logrank.pval = model$pvalue)}) %>%
  bind_rows(.id = "Variable") %>%
  arrange(logrank.pval)

logrank_results %>%
  mutate(logrank.pval = ifelse(logrank.pval < 0.001, "<0.001", round(logrank.pval,4)))
## Univariate proportional hazards models

# Fit PH models from baseline variables
model_list <- list(
  # Initial disease status
  disgroup = coxph(dfs_surv ~ disgroup, bmt_agvhd),
  fab = coxph(dfs_surv ~ fab, bmt_agvhd),
  waittime = coxph(dfs_surv ~ waittime, bmt_agvhd),
  mtx = coxph(dfs_surv ~ mtx, bmt_agvhd),
  hospital = coxph(dfs_surv ~ hospital, bmt_agvhd),
```

```
  # Patient
  age = coxph(dfs_surv ~ age, bmt_agvhd),
  male = coxph(dfs_surv ~ male, bmt_agvhd),
  cmv = coxph(dfs_surv ~ cmv, bmt_agvhd),
  # Donor
  donorage = coxph(dfs_surv ~ donorage, bmt_agvhd),
  donormale = coxph(dfs_surv ~ donormale, bmt_agvhd),
  donorcmv = coxph(dfs_surv ~ donorcmv, bmt_agvhd)
)

# Extract results from each model
results <- lapply(model_list, function(model) {
  tidymodel <- broom::tidy(model, conf.int = TRUE, conf.level = 0.9)

  data.frame(
    HR = exp(tidymodel$estimate),
    HR.low = exp(tidymodel$conf.low),
    HR.high = exp(tidymodel$conf.high),
    logrank.pval = broom::glance(model)$p.value.sc[[1]], # Log-rank test
    LRT.pval = tidymodel$p.value  # Likelihood ratio test (LRT)
    )
  }
)
cox_results <- bind_rows(results, .id = "Model")

# Forest plot of HR estimates
ggplot(cox_results, aes(x=HR, y=Model, color=Model)) +
  geom_point(size = 3) + # Plot hazard ratio points
  geom_errorbarh(aes(xmin=HR.low, xmax=HR.high), height=0.2) +  # CI bars
  geom_vline(xintercept=1, linetype="dashed", color="red") +  # Reference line
  labs(title = "Forest Plot from Univariate Models",
       x = "Hazard Ratio (HR)",
       y = "Variable") +
  theme(legend.position = "bottom")

## Everything PH model

# Fit a single PH model from baseline variables
results <- coxph(dfs_surv ~
        # Initial disease status
        disgroup + fab + waittime + mtx + hospital +
        # Patient
        age + male + cmv +
        # Donor
        donorage + donormale + donorcmv,
      bmt_agvhd) %>%
  tidy(conf.int = TRUE, conf.level = 0.9)

# Extract results
cox_results <- results %>%
  mutate(Variable = term,
         HR = exp(estimate),
         HR.low = exp(conf.low),
```

```
         HR.high = exp(conf.high),
         LRT.pval = p.value,
         .keep="none")

# Forest plot of HR estimates
ggplot(cox_results, aes(x=HR, y=Variable, color=Variable)) +
  geom_point(size = 3) + # Plot hazard ratio points
  geom_errorbarh(aes(xmin=HR.low, xmax=HR.high), height=0.2) +  # CI bars
  geom_vline(xintercept=1, linetype="dashed", color="red") +  # Reference line
  labs(title = "Forest Plot from Everything Model",
       x = "Hazard Ratio (HR)",
       y = "Variable") +
  theme(legend.position = "bottom")

## Everything model stratified by hospital

# Fit a single PH model from baseline variables
results <- coxph(dfs_surv ~ disgroup + fab + waittime + mtx + strata(hospital) +
                   age + male + cmv + donorage + donormale + donorcmv,
                 bmt_agvhd) %>%
  tidy(conf.int = TRUE, conf.level = 0.9)

# Extract results
cox_results <- results %>%
  mutate(Variable = term, HR = exp(estimate), HR.low = exp(conf.low),
         HR.high = exp(conf.high), LRT.pval = p.value, .keep="none")

# Forest plot of HR estimates
ggplot(cox_results, aes(x=HR, y=Variable, color=Variable)) +
  geom_point(size = 3) + # Plot hazard ratio points
  geom_errorbarh(aes(xmin=HR.low, xmax=HR.high), height=0.2) +  # CI bars
  geom_vline(xintercept=1, linetype="dashed", color="red") +  # Reference line
  labs(title = "Forest Plot from Everything Model with hospital stratification",
       x = "Hazard Ratio (HR)",
       y = "Variable") +
  theme(legend.position = "bottom")
```

End of document.