# Experiment 2

A & S

2023-08-21

```r
max_s_opts = c(5, 10, 20, 30, 40)
nfolds = 5

feature_df <- data.frame()

weight_opts_literature = c(0.1, seq(10, 110, 10))
```

```r
# train UBayFS models to obtain features
max_s_opts = 20

system.time(
for(weight in weight_opts_literature){
  for(max_s in max_s_opts) {
    set.seed(1)
    for (test_fold in 1:nfolds) {
      weight_vec = rep(0.1, nfeats)
      weight_vec[prior_features_all_ind] = weight

      block_const <- UBayFS::buildConstraints(constraint_types = c("max_size"),
                                              constraint_vars = max_s,
                                              num_elements = nfeats,
                                              rho = Inf)

      model = train_UBay_model(data = data_list[[test_fold]]$train_data,
                               target = as.numeric(data_list[[test_fold]]$train_target$category),
                               M = 100,
                               tt_split = 0.75,
                               nf = max_s,
                               weights = weight_vec,
                               lam = 10,
                               block_constraints = list(block_const))

      features = which(model$output$feature_set[1,] == 1)
      feature_df <- rbind(feature_df, data.frame(index = features,
                                                 max_s = max_s,
                                                 fold = test_fold,
                                                 weight = weight))
    }
  }
}
)
```

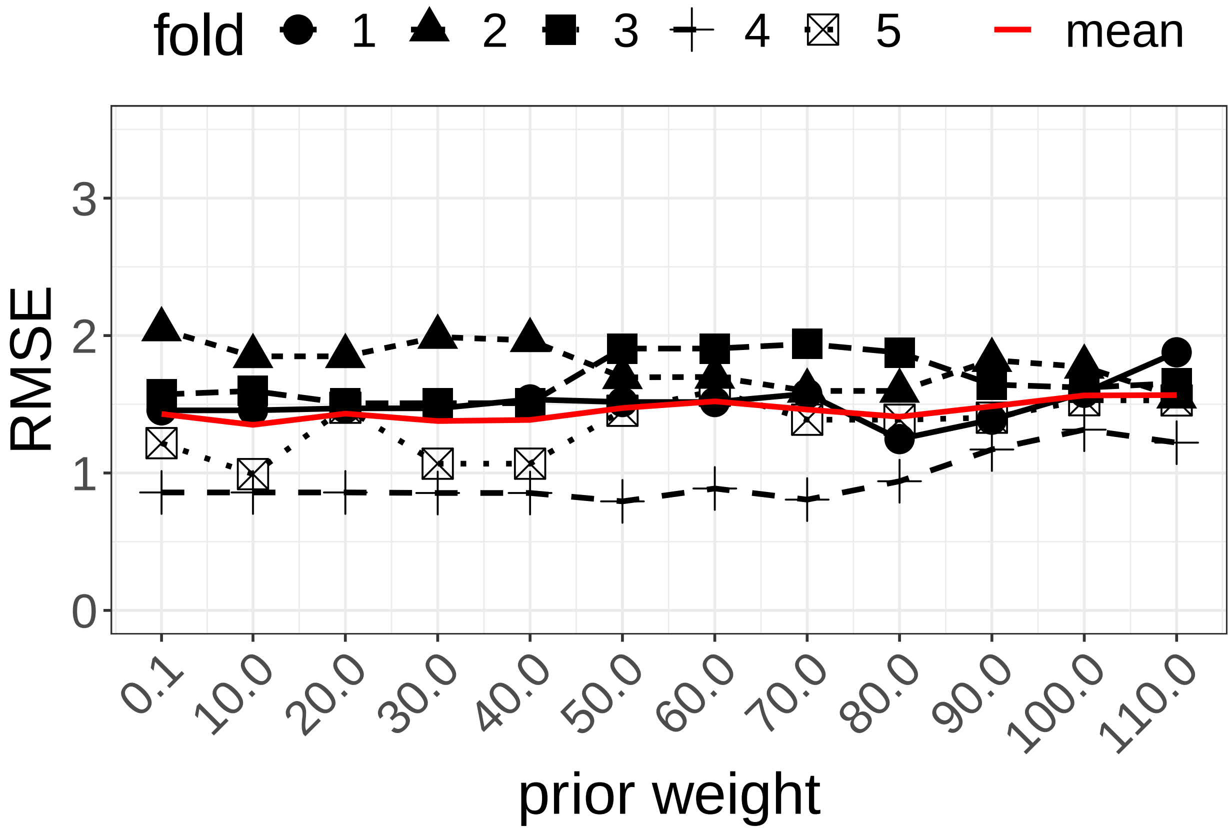
```
##        User      System verstrichen
##       52.28        1.60      152.46
```

```
save(feature_df, file = paste0(path, "data/exp2_features.Rdata"))
```

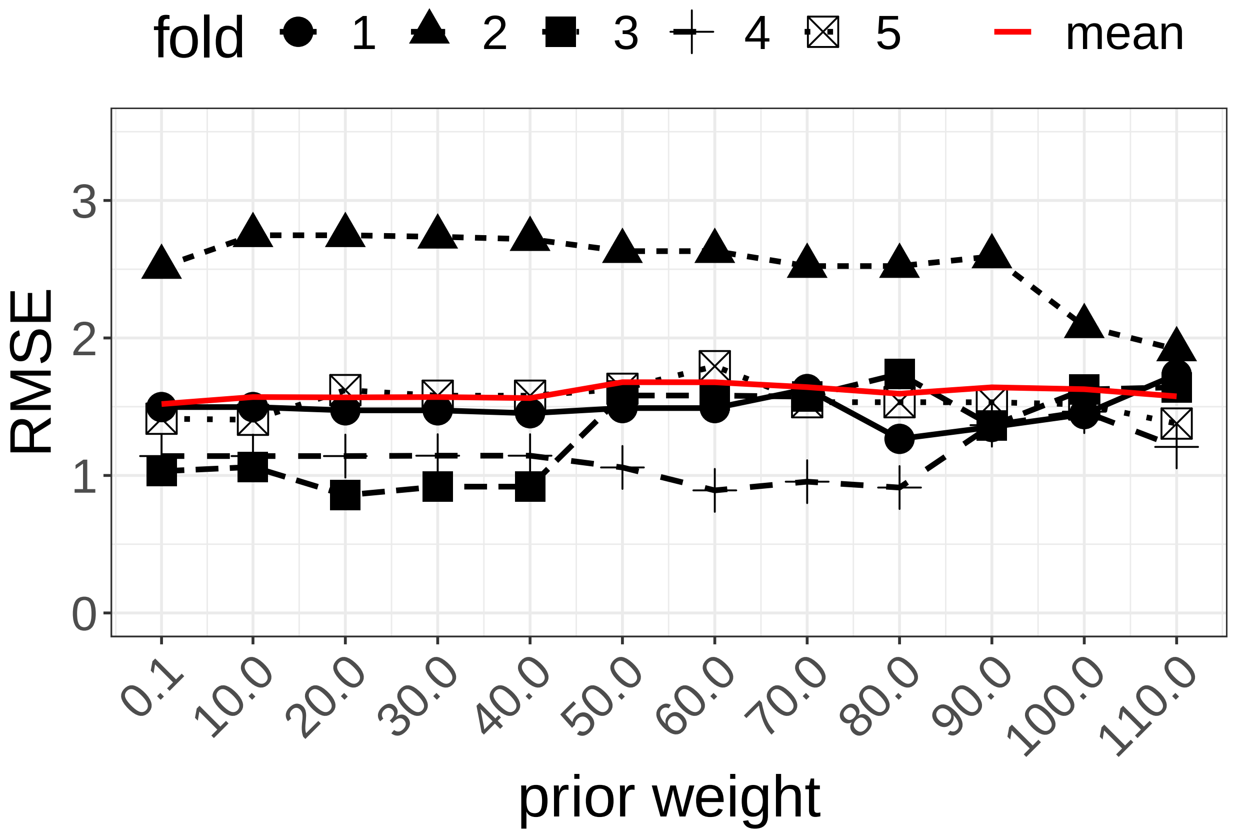
```
max_s_opts = 20

pred_df <- data.frame()
RED_df <- data.frame()
stability_df <- data.frame()
param_df <- data.frame()

# signs within elementary models
feature_df <- cbind(feature_df, sign = 0)

# PREDICTIONS
for(weight_ in weight_opts_literature){
  for (max_s_ in max_s_opts) {
    set.seed(1)
    for (test_fold in 1:nfolds) {

      features = subset(feature_df, (max_s == max_s_) & (fold == test_fold) & (weight == weight_))$inde
      eval <- eval_all(features,
                                train_data = data_list[[test_fold]]$train_data,
                                test_data = data_list[[test_fold]]$test_data,
                                train_target = data_list[[test_fold]]$train_target,
                                test_target = data_list[[test_fold]]$test_target#,
                       #fixed_param = 3)
      )

      pred_df <- rbind(pred_df,
                       cbind(eval$res,
                             max_s = max_s_,
                             fold = test_fold,
                             weight = weight_))

      param_df <- rbind(param_df,
                    data.frame(param = eval$param,
                               method = names(eval$param),
                               max_s = max_s_,
                               fold = test_fold,
                               weight = weight_))

      feat_inds <- (feature_df$index %in% features &
                      feature_df$max_s == max_s_ &
                      feature_df$fold == test_fold &
                      feature_df$weight == weight_)
      feature_df[feat_inds, "sign"] = feature_df[feat_inds, "sign"] + eval$sign

      # RED
      cor_mat <- cor(rbind(data_list[[test_fold]]$train_data[, features],
                           data_list[[test_fold]]$test_data[, features])) # feature-wise correlation ma
      RED_df <- rbind(RED_df,
                      data.frame(RED = mean(abs(cor_mat[upper.tri(cor_mat)])),
                                 max_s = max_s_,
                                 fold = test_fold,
                                 weight = weight_))
    }

    # stability
    f_mat <- matrix(0, nrow = nfeats, ncol = nfolds)
    features <- subset(feature_df, (max_s == max_s_) & (weight == weight_)) %>%
      select(index, fold) %>%
      as.matrix()
    f_mat[features] <- f_mat[features] + 1

    stab = getStability(t(f_mat))
    stability_df <- rbind(stability_df,
                          cbind(max_s = max_s_,
                                weight = weight_,
                                stab))
  }
}

RED_df <- RED_df %>% group_by(max_s, weight) %>%
  summarize(value = mean(RED),
            variance = var(RED),
            lower = min(RED),
            upper = max(RED)) %>%
  as.data.frame()

summary_df <- pred_df %>% group_by(metric, model, type, max_s, weight) %>%
  summarize(mean = mean(value),
            median = median(value),
            sd = sd(value),
            min = min(value),
            max = max(value)) %>%
  as.data.frame()

save(feature_df, pred_df, RED_df, stability_df, summary_df, param_df,
     file = paste0(path, "data/exp2_predictions.Rdata"))
```

# PREDICTION PLOT UBAYFS

```
load(paste0(path, "data/exp2_features.Rdata"))
load(paste0(path, "data/exp2_predictions.Rdata"))

plot_performance2 <- function(type_ = "test", metric_ = "RMSE", model_ = "knn", max_s_ = 20){
  p <- pred_df %>% subset(type == type_ & metric == metric_ & model == model_ & max_s == max_s_)  %>%
    mutate(fold = as.factor(fold)) %>%
    ggplot(aes(x = weight)) +
    geom_point(aes(y = value, pch = fold), size = 5) +
    geom_line(aes(y = value, lty = fold), linewidth = 1) +
    geom_line(data = summary_df %>% subset(max_s == max_s_ & type == type_ & metric == metric_ & model
              aes(x = weight,
                  y = mean,
                  color = "mean"),
              linewidth = 1) +
    ylim(0, 3.5)+
    ylab("RMSE") +
    xlab("prior weight") +
    scale_x_continuous(breaks = weight_opts_literature) +
    theme_bw() +
    theme(text = element_text(size = 22),
          legend.position = "top",
          axis.text.x = element_text(angle = 45, hjust = 1)) +
    guides(fill="none", color = guide_legend(title = "")) +
    scale_fill_manual(values = "red") +
    scale_color_manual(values = "red")

  return(p)
}

p <- plot_performance2(max_s_ = 20, type_ = "test", metric_ = "RMSE", model_ = "knn")
p
```



```
ggsave(p, filename = paste0(path, "plots/experiment_2_performance_max_s_kNN_UBayFS.png"), width = 8, he

p <- plot_performance2(max_s_ = 20, type_ = "test", metric_ = "RMSE", model_ = "linear")
p
```



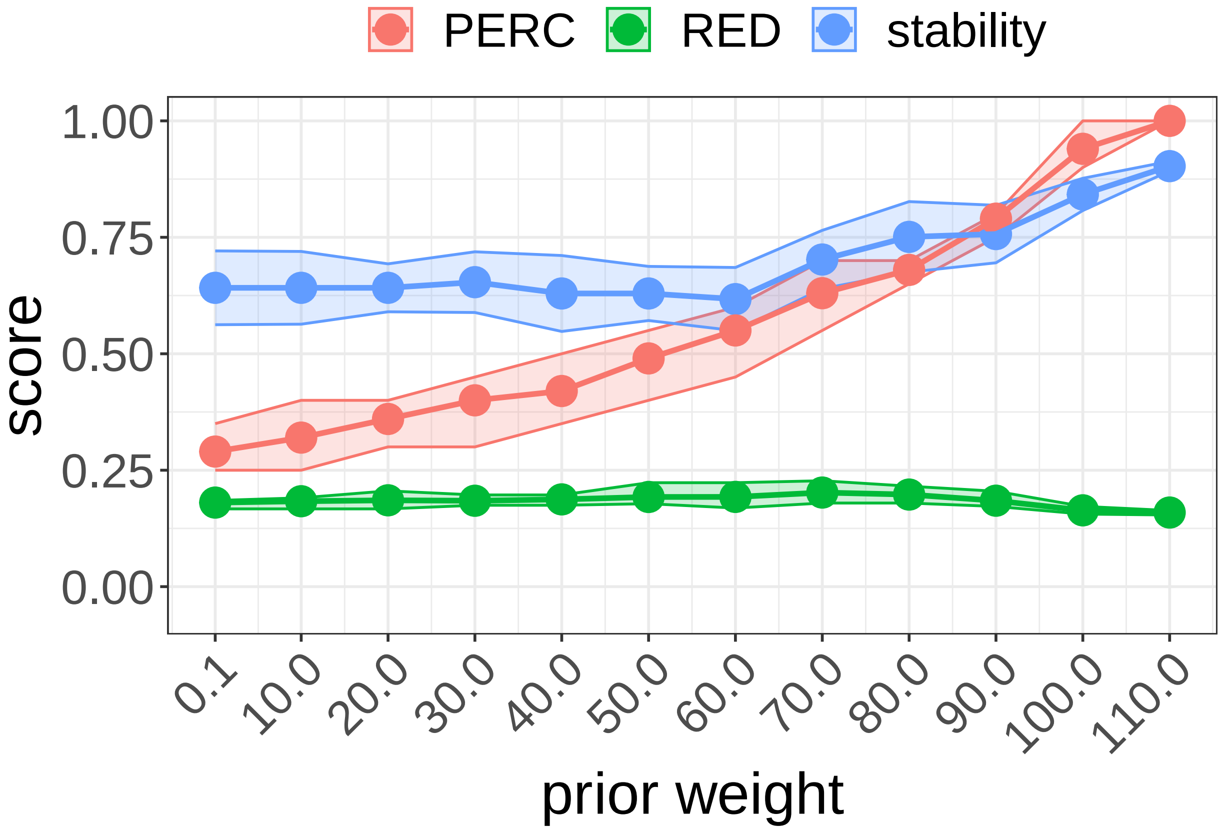
```
ggsave(p, filename = paste0(path, "plots/experiment_2_performance_max_s_linear_UBayFS.png"), width = 8,
```

### Check how often the prior features were selected.

```
PERC_df <- feature_df %>% subset(max_s == 20) %>%
  group_by(fold, weight) %>%
  summarize(num_abs = mean(index %in% prior_features_all_ind)) %>%
  group_by(weight) %>%
  summarize(value = mean(num_abs),
            variance = var(num_abs),
            lower = min(num_abs),
            upper = max(num_abs))
```

## STABILITY PLOT UBAYFS

```
plot_stability2 <- function(max_s_ = 20){
  p <- rbind(cbind(stability_df, score = "stability"),
             cbind(RED_df, score = "RED"),
             cbind(max_s = 20,
                   PERC_df,
                    score = "PERC")
               ) %>%
    subset(max_s == max_s_)  %>%
    ggplot(aes(x = weight, y = value, color = score, fill = score)) +
    geom_ribbon(aes(ymin = lower, ymax = upper), alpha = 0.2)+
    geom_point(size = 5) +
    geom_line(linewidth = 1) +
    ylim(-0.05, 1)+
    ylab("score") +
    xlab("prior weight") +
    scale_x_continuous(breaks = weight_opts_literature) +
    theme_bw() +
    theme(text = element_text(size = 22),
          legend.position = "top",
          legend.title = element_blank(),
          axis.text.x = element_text(angle = 45, hjust = 1))

  return(p)
}

p <- plot_stability2(max_s_ = 20)
p
```



```
ggsave(p, filename = paste0(path, "plots/experiment_2_stability_red.png"), width = 8, height = 5)
```

# FEATURE TABLE

```r
load(file = paste0(path, "data/feature_counts_exp1.Rdata"))

# original order: blood, image, hist, NA, pat_char, treatment;
# new order: pat_char, blood, hist, image, treatment
block_reordering <- c("1" = "2", "2" = "4","3" = "3","5" = "1","6" = "5")
block_index_new <- block_index %>% as.character() %>% plyr::revalue(replace = block_reordering) %>% as.
feature_order <- order(block_index_new)

weight_levels <- c("50", "110")
f_mat <- matrix(0, ncol = 2, nrow = nfeats)
sign_mat <- matrix(0, ncol = 2, nrow = nfeats)
colnames(f_mat) <- colnames(sign_mat) <- weight_levels

for(fold_ in 1:nfolds){
  for(weight_ in weight_levels){
    features <- feature_df %>% subset(max_s == 20 & weight == weight_ & fold == fold_)
    f_mat[features$index, weight_] <- f_mat[features$index, weight_] + 1
    sign_mat[features$index, weight_] <- sign_mat[features$index, weight_] + features$sign
  }
}

weight_cols = data.frame(
  UBayFS_with_prior50 = paste0(f_mat[,"50"],
                                  ifelse(sign_mat[,"50"] != 0, paste0("(",
                                         ifelse(sign_mat[,"50"] > 0,
                                                ifelse(sign_mat[,"50"] == f_mat[,"50"], "++", "+"),
                                                ifelse(sign_mat[,"50"] == -f_mat[,"50"], "--", "-")),
                                                ")"), "")),
  UBayFS_with_prior110 = paste0(f_mat[,"110"],
                                  ifelse(sign_mat[,"110"] != 0, paste0("(",
                                         ifelse(sign_mat[,"110"] > 0,
                                                ifelse(sign_mat[,"110"] == f_mat[,"110"], "++", "+"),
                                                ifelse(sign_mat[,"110"] == -f_mat[,"110"], "--", "-")),
                                                ")"), ""))
  )

feature_counts <- cbind(feature_counts,
                        weight_cols)

feature_counts[feature_order,] %>% save(file = paste0(path, "data/feature_counts_exp2.Rdata"))
print(xtable(feature_counts[feature_order,], display = c("s","s","s","s","s","s", "s", "s")), include.r
```

```
## % latex table generated in R 4.2.2 by xtable 1.8-4 package
## % Mon Aug 21 09:51:24 2023
## \begin{table}[ht]
## \centering
## \begin{tabular}{llllllll}
##   \hline
## block & prior & name & RENT & UBayFS & UBayFS\_with\_prior50 & UBayFS\_with\_prior110 \\
##   \hline
##  & * & Age at Diagnosis & 2 & 1(--) & 5(-) & 5(--) \\
##    &  & Time from PET to Metastasis (days)  & 0 & 1(++) & 1(++) & 0 \\
##    &  & Time from PET to Diagnosis  (days)  & 0 & 0 & 0 & 0 \\
##    &  & Time from diag to mets  (months)  & 0 & 0 & 0 & 0 \\
##    &  & Sex & 0 & 0 & 0 & 0 \\
##    &  & Loc. Adv. Resectable Disease & 0 & 0 & 0 & 0 \\
##    &  & Loc. Reccurence & 0 & 0 & 0 & 0 \\
##    &  & Metastatic Disease at Time of Diagnosis & 3(+) & 1(++) & 0 & 0 \\
##    &  & Treatment Intention Palliative & 4(-) & 5(--) & 3(--) & 0 \\
##    &  & Prior Other Cancer & 2(++) & 1(--) & 0 & 0 \\
##    &  & Living Alone & 0 & 0 & 0 & 0 \\
##    & * & TNM staging Pathological & 0 & 0 & 0 & 4(--) \\
##    &  & Stage grouped Stage IV & 0 & 0 & 0 & 0 \\
##    &  & Mets Bone & 5(--) & 5(--) & 5(--) & 0 \\
##    &  & Mets LN Distant & 0 & 0 & 0 & 0 \\
##    &  & Mets LN Regional & 0 & 0 & 0 & 0 \\
##    &  & Mets LN Retro & 0 & 0 & 0 & 0 \\
##    &  & Mets LN & 0 & 0 & 0 & 0 \\
##    &  & Mets Liver & 0 & 0 & 0 & 0 \\
##    &  & Mets Lung & 0 & 0 & 0 & 0 \\
##    &  & Mets Other & 0 & 0 & 0 & 0 \\
##    &  & Mets Skin & 0 & 0 & 0 & 0 \\
##    &  & Primary Tumour Resected & 0 & 0 & 0 & 0 \\
##    &  & M-stage M1 & 0 & 0 & 0 & 0 \\
##    &  & BMI & 1(--) & 0 & 0 & 0 \\
##    &  & Non Smoker & 0 & 0 & 0 & 0 \\
##    &  & Smoker & 0 & 0 & 0 & 0 \\
##    &  & Radical Surgery & 3(++) & 4 & 4(+) & 0 \\
##    &  & Co-morbidity Severity 1 & 0 & 0 & 0 & 0 \\
##    &  & Co-morbidity Severity  $>$ 1 & 0 & 0 & 0 & 0 \\
##    &  & T-stage T2 & 0 & 0 & 0 & 0 \\
##    &  & T-stage T3 & 0 & 0 & 0 & 0 \\
##    &  & T-stage T4 & 2(--) & 2(--) & 1(--) & 0 \\
##    &  & N-stage N1 & 0 & 0 & 0 & 0 \\
##    &  & N-stage $>$  N1 & 0 & 0 & 0 & 0 \\
##    & * & WHO Perf Stat 1 & 0 & 0 & 3(--) & 5(+) \\
##    & * & WHO Perf Stat 2 & 4(--) & 5(--) & 5(--) & 5(--) \\
##    & * & WHO Perf Stat 3 & 0 & 0 & 0 & 3(--) \\
##    & * & WHO Perf Stat 4 & 0 & 0 & 0 & 2(-) \\
##    &  & Abs.  Neutrophil Count & 0 & 0 & 0 & 0 \\
##    & * & Albumin & 2 & 4(--) & 5(-) & 5(++) \\
##    &  & CRP & 5(-) & 5(--) & 5(-) & 0 \\
##    &  & Creatinine & 0 & 0 & 0 & 0 \\
##    &  & Haemoglobin & 0 & 0 & 0 & 0 \\
##    &  & WBC & 1(--) & 1(--) & 1(--) & 0 \\
##    &  & ALP $>$ Normal $<$= 3UNL & 4(--) & 5(--) & 4(--) & 0 \\
##    &  & ALP $>$ 3UNL & 1(++) & 0 & 0 & 0 \\
##    &  & Chromogranin\_A $>$ Normal $<$= 2UNL & 0 & 0 & 0 & 0 \\
##    &  & Chromogranin\_A $>$ 2UNL & 0 & 0 & 0 & 0 \\
##    & * & LDH $>$ Normal  $<$= 2UNL & 0 & 0 & 1(++) & 4(++) \\
##    & * & LDH $>$ 2UNL & 0 & 0 & 1(++) & 5(+) \\
##    &  & NSE $>$ Normal $<$= 2UNL & 0 & 0 & 0 & 0 \\
##    &  & NSE $>$ 2UNL & 0 & 0 & 0 & 0 \\
##    & * & Platelets & 2(--) & 5(--) & 5(--) & 5(--) \\
##    & * & Ki-67 & 5(--) & 5(--) & 5(--) & 5(--) \\
##    &  & Hist Exam Metastasis & 0 & 0 & 0 & 0 \\
##    & * & Primary Tumour Esophagus & 0 & 0 & 0 & 4 \\
##    & * & Primary Tumour Gallbladder/duct & 0 & 0 & 1(++) & 5(-) \\
##    & * & Primary Tumour Gastric & 0 & 0 & 1(--) & 5(-) \\
##    & * & Primary Tumour Other abdominal & 0 & 0 & 0 & 4(-) \\
##    & * & Primary Tumour Pancreas & 1(++) & 0 & 1(++) & 5(+) \\
##    & * & Primary Tumour Rectum & 0 & 0 & 0 & 5(++) \\
##    & * & Unknown Pr. With Dominance of GI met. & 0 & 0 & 0 & 4 \\
##    &  & Co-existing Neoplasm Adenoma & 0 & 0 & 0 & 0 \\
##    &  & Co-existing Neoplasm Dysplasia & 0 & 0 & 0 & 0 \\
##    &  & No Co-existing Neoplasm & 0 & 0 & 0 & 0 \\
##    & * & Tumour Morphology WD & 4(+) & 3(--) & 4 & 5(--) \\
##    &  & Chromogranin A Staining & 0 & 0 & 0 & 0 \\
##    &  & Architecture Infiltrative & 1(++) & 0 & 0 & 0 \\
##    &  & Architecture Organoid & 1(++) & 1(++) & 0 & 0 \\
##    &  & Architecture Solid & 0 & 0 & 0 & 0 \\
##    &  & Architecture Trabecular & 1(--) & 1(++) & 0 & 0 \\
##    &  & Vessel Pattern Distant & 1(++) & 2(--) & 1(--) & 0 \\
##    &  & Biopsy Location Gastric & 0 & 0 & 0 & 0 \\
##    &  & Biopsy Location Liver Metastasis & 0 & 0 & 0 & 0 \\
##    &  & Biopsy Location Lymph Node & 0 & 0 & 0 & 0 \\
##    &  & Biopsy Location  Oesophagus & 0 & 0 & 0 & 0 \\
##    &  & Biopsy Location Pancreas & 0 & 0 & 0 & 0 \\
##    &  & Biopsy Location Peritoneum & 2 & 1(--) & 0 & 0 \\
##    &  & No Stroma & 4(++) & 1(++) & 1(++) & 0 \\
##    &  & Stroma & 3(++) & 3(-) & 2(--) & 0 \\
##    &  & Geographic Necrosis & 0 & 1(--) & 0 & 0 \\
##    &  & Synaptophysin Staining 2+  & 0 & 0 & 0 & 0 \\
##    &  & Synaptophysin Staining 3+ & 0 & 0 & 0 & 0 \\
##    &  & Injection to Scan [min] & 2 & 2(++) & 2(++) & 0 \\
##    &  & Weight [kg] & 2(--) & 0 & 0 & 0 \\
##    & * & Total MTV [cm^3] & 3(+) & 1(--) & 4(++) & 5(--) \\
##    &  & SUVmean & 0 & 0 & 0 & 0 \\
##    & * & SUVmax & 2 & 4 & 5(++) & 5(-) \\
##    &  & SUVmean (total) & 1(++) & 0 & 0 & 0 \\
##    &  & SUVmax (total) & 5(--) & 5(--) & 5(--) & 0 \\
##    & * & Total TLG [g] & 4 & 1(++) & 3(+) & 5(+) \\
##    &  & Institution Rikshospitalet & 4(++) & 4(++) & 2(++) & 0 \\
##    &  & Institution Ullevaall & 0 & 0 & 0 & 0 \\
##    &  & Height [cm] & 0 & 0 & 0 & 0 \\
##    &  & Glucose [mmol/L] & 2(--) & 0 & 0 & 0 \\
##    &  & Time from PET to first treatment  (days)  & 0 & 0 & 0 & 0 \\
##    &  & Chemotherapy Type Cisplatin/Etoposide & 4(+) & 3(+) & 2(--) & 0 \\
##    &  & Chemotherapy Type Other & 0 & 0 & 0 & 0 \\
##    &  & Chemotherapy Type Temozolomide/Capecitabine & 1(++) & 0 & 0 & 0 \\
##    &  & Chemotherapy Type Temozolomide/Everolimus & 4(++) & 5(++) & 4(++) & 0 \\
##    &  & Best Response  (RECIST) Not Assessed & 0 & 0 & 0 & 0 \\
##    &  & Best Response  (RECIST) Only Clinical PD & 0 & 0 & 0 & 0 \\
##    &  & Best Response  (RECIST) Partial Response & 2(--) & 0 & 0 & 0 \\
##    &  & Best Response  (RECIST) Progressive Disease  & 0 & 0 & 0 & 0 \\
##    &  & Best Response  (RECIST) Stable Disease  & 0 & 0 & 0 & 0 \\
##    &  & Reintroduction with Cisplatin Etoposide & 0 & 0 & 0 & 0 \\
##    &  & Number of Courses & 4(++) & 4(++) & 3(++) & 0 \\
##    &  & Treatment Stopped Other & 1(++) & 1(++) & 1(++) & 0 \\
##    &  & Treatment Stopped Progression of Disease & 0 & 0 & 0 & 0 \\
##    &  & Treatment Stopped Toxicity & 0 & 0 & 0 & 0 \\
##    &  & No Progression & 5(++) & 4(++) & 3(++) & 0 \\
##    &  & Progression & 3 & 3(+) & 1(--) & 0 \\
##    \hline
## \end{tabular}
## \end{table}
```