Supplementary Troponin Analysis

Improving risk stratification for patients with type 2 myocardial infarction

The objective of this analysis is to construct a linear regression model, which permits to predict log troponin I from log tropinin T. Such model, will permit to employ the proposed T-2 risk score when we only have access to troponin T.

Data pre-processing

The data here used **comes from ADD HERE INFO ABOUT SUBSTUDY**. The dataset contains two readings of both troponin I and troponin T of 1869 patients. In addition, the dataset contains an adjudication code, where:

- adj = 1 corresponds to Type 1 Myocardial infarction
- adj = 2 corresponds to Type 2 Myocardial infarction
- adj = 3 corresponds to Myocardial injury
- adj = 9 corresponds to NA
- adj = NA corresponds to No injury

We load the dataset and make the above adjudication codes explicit:

```r
library(readr)
substudy <-  as.data.frame(read_csv("~/Documents/Postdoc/DEMAND/highsteacs_substudy_troponin.csv"))

str(substudy)
```

```
## 'data.frame':    1869 obs. of  6 variables:
##  $ substudyid : num  1 2 3 5 6 7 8 9 10 11 ...
##  $ tni1_result: num  11508 3 5 3 8 ...
##  $ tni2_result: num  15733 4 5 3 12 ...
##  $ tnt1_result: num  712 4 17 7 6 11 4.99 66 12 20 ...
##  $ tnt2_result: num  NA 6 17 6 7 11 4.99 62 14 25 ...
##  $ adj        : num  1 NA NA NA NA NA NA 2 NA 1 ...
```

```r
substudy$adj[substudy$adj == 1] <- "Type 1 MI"
substudy$adj[substudy$adj == 2] <- "Type 2 MI"
substudy$adj[substudy$adj == 3] <- "Myocardial injury"
substudy$adj[substudy$adj == 9] <- NA
substudy$adj[is.na(substudy$adj)==TRUE] <- "No injury"
```

Because our aim is to model the relationship between tropinin I and tropinin T. Below, we re-arrange the data by stacking the two available tropinin readings. In addition, we remove rows of the stacked dataset where at leat one of the troponin reading were unvailable.

```r
stacked_data <- data.frame(cbind( tni = c(substudy$tni1_result, substudy$tni2_result),
                                  tnt = c(substudy$tnt1_result, substudy$tnt2_result)),
                                  adj = c(substudy$adj, substudy$adj))
```

```r
#remove NAs
NA.I <- which(is.na(stacked_data$tni)==TRUE)
NA.T <- which(is.na(stacked_data$tnt)==TRUE)
stacked_data <- stacked_data[-c(unique(c(NA.I, NA.T))),]
row.names(stacked_data) <- NULL

#Number of available readings
nrow(stacked_data)
```

```
## [1] 3559
```

We further remove any troponin readings above and below the limit of detection of the assays employed (“ARCHITECT Stat High Sensitivity Troponin-I”)

- Lower limit of detection for troponin I is 3.5 ng/L and for troponin T 6.0 ng/L.
- Upper limit of detection for troponin I is 5,000 ng/L and for troponin T 10,000 ng/L.

```r
below_limit <- unique(c(which(stacked_data$tni <= 3.5), which(stacked_data$tnt <= 6)))
above_limit <- unique(c(which(stacked_data$tni >= 5000), which(stacked_data$tnt >= 10000)))

data_LOD <- stacked_data[-c(below_limit, above_limit),]
row.names(data_LOD) <- NULL
nrow(data_LOD)
```

```
## [1] 1327
```

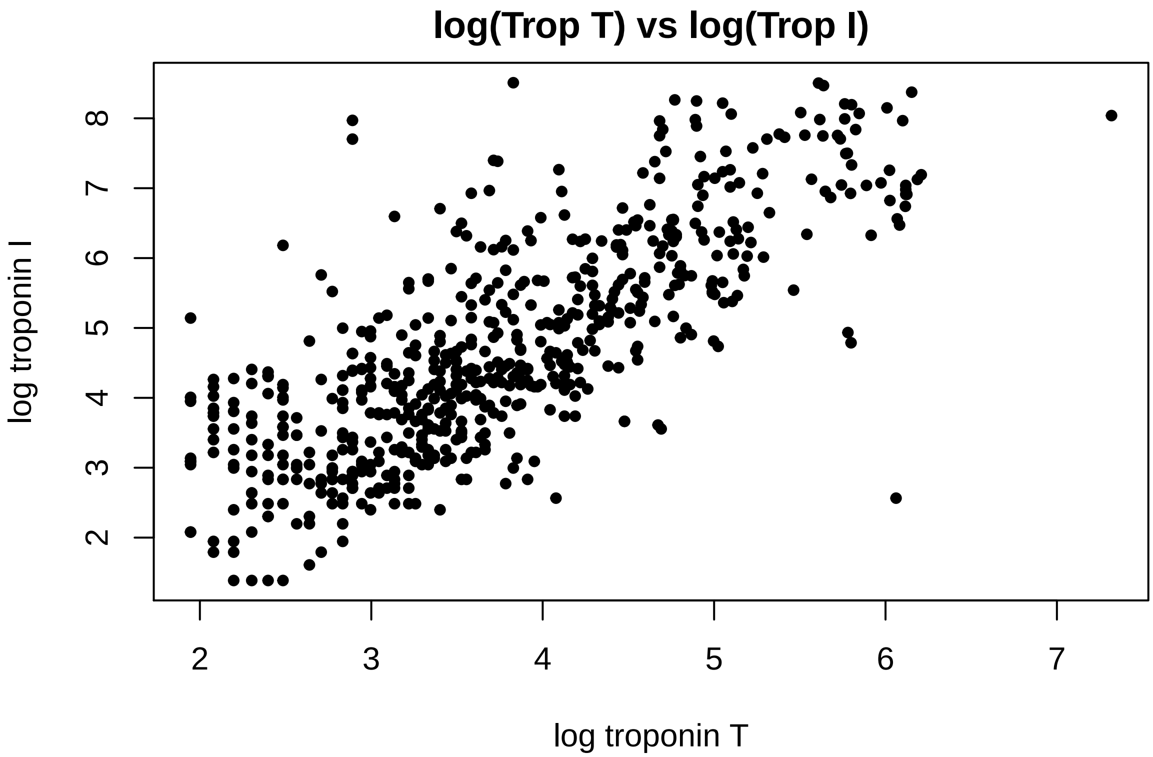
Furthermore, as our objective is to model the relationship between troponin I and T in populations with MI, we remove readings corresponding to those subjects that have an adjudicated diagnose of no injury.

```r
data_LOD_subset <- data_LOD[-which(data_LOD$adj == "No injury"),]
row.names(data_LOD_subset) <- NULL

#Number of available troponin readings
nrow(data_LOD_subset)
```

```
## [1] 619
```

Finally, we compute the logarithm of both troponin I and T and produce a scatter plot of the data:

```r
data_LOD_subset$log.TropI <- log(data_LOD_subset$tni)
data_LOD_subset$log.TropT <- log(data_LOD_subset$tnt)

par(mar = c(4, 4, 2, .1))
plot(data_LOD_subset$log.TropT, data_LOD_subset$log.TropI,
     pch = 20,
     main = "log(Trop T) vs log(Trop I)",
     xlab = "log troponin T", ylab = "log troponin I")
```



Model fitting

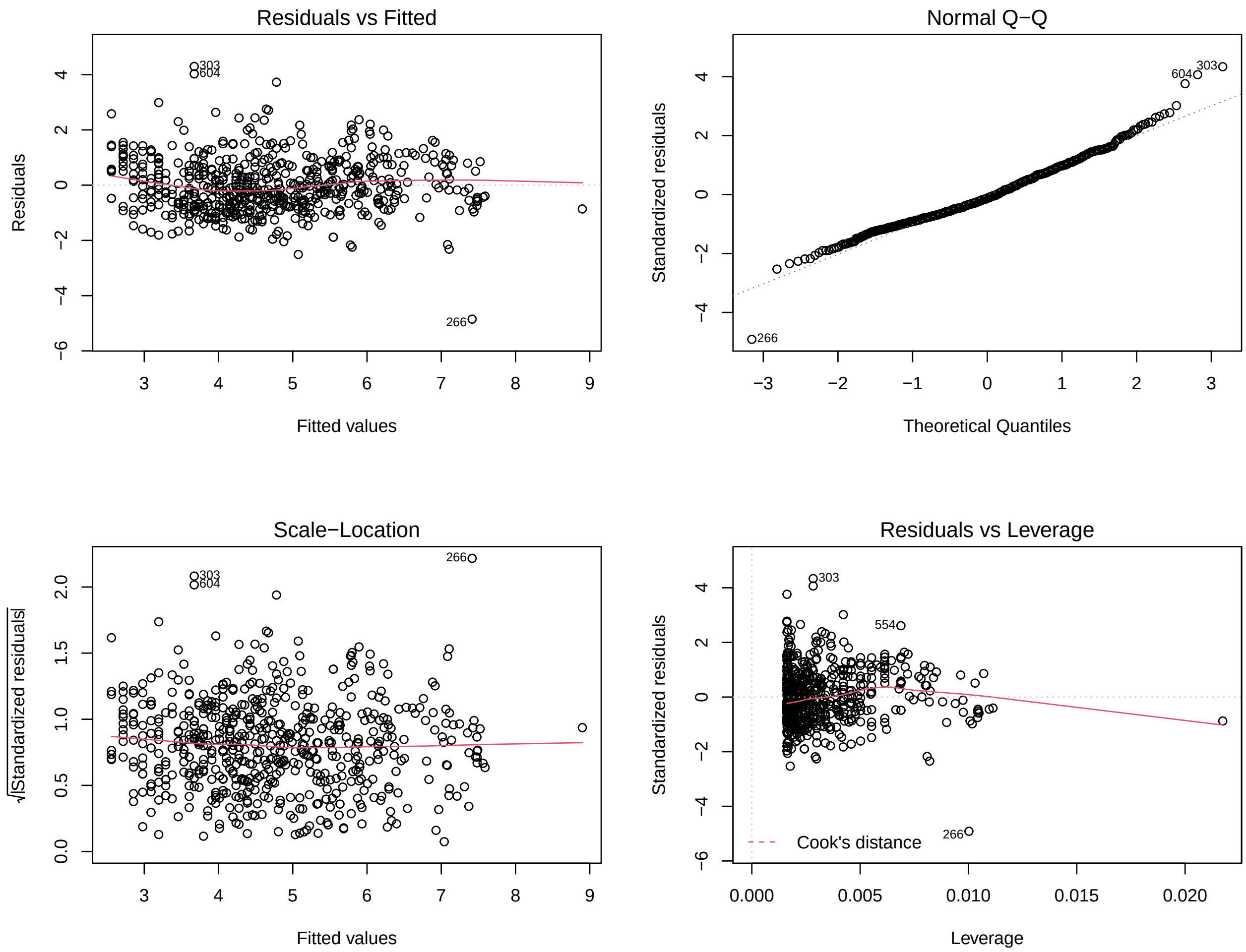
We fit a linear regression model:

```
trop.fit <- lm( log.TropI ~ log.TropT, data = data_LOD_subset)
summary(trop.fit)
```

```
##
## Call:
## lm(formula = log.TropI ~ log.TropT, data = data_LOD_subset)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.8513 -0.6859 -0.1272  0.6591  4.2974
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.2612     0.1543   1.692   0.0911 .
## log.TropT     1.1804     0.0396  29.807   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9923 on 617 degrees of freedom
## Multiple R-squared:  0.5902, Adjusted R-squared:  0.5895
## F-statistic: 888.5 on 1 and 617 DF,  p-value: < 2.2e-16
```

And produce residuals plots:

```
par(mfrow=c(2,2))
plot(trop.fit)
```



From the plots above, we remove all strong outliers. In addition, we also remove one observation with an extremely high value in troponin T and one with an extremely high value in tropinin I, which are not inline with the overall trend in the data. The removed observations ar shown below:

```
data_LOD_subset[c(303,604,266,554,444,110,422,
                  which(data_LOD_subset$log.TropT == max(data_LOD_subset$log.TropT)),
                  which(data_LOD_subset$log.TropI == max(data_LOD_subset$log.TropI))),]
```

```
##      tni  tnt               adj log.TropI log.TropT
## 303 2894   18 Myocardial injury  7.970395  2.890372
## 604 2213   18 Myocardial injury  7.702104  2.890372
## 266   13  429          Type 1 MI  2.564949  6.061457
## 554  171    7          Type 1 MI  5.141664  1.945910
## 444  484   12          Type 1 MI  6.182085  2.484907
## 110 1612   42          Type 1 MI  7.385231  3.737670
## 422 1633   41          Type 1 MI  7.398174  3.713572
## 50  3100 1508 Myocardial injury  8.039157  7.318540
## 614 4961   46          Type 1 MI  8.509363  3.828641
```

```
data_LOD_subset2 <-  data_LOD_subset[-c(303,604,266,554,444,110,422,
                                        which(data_LOD_subset$log.TropT ==max(data_LOD_subset$log.TropT)
```

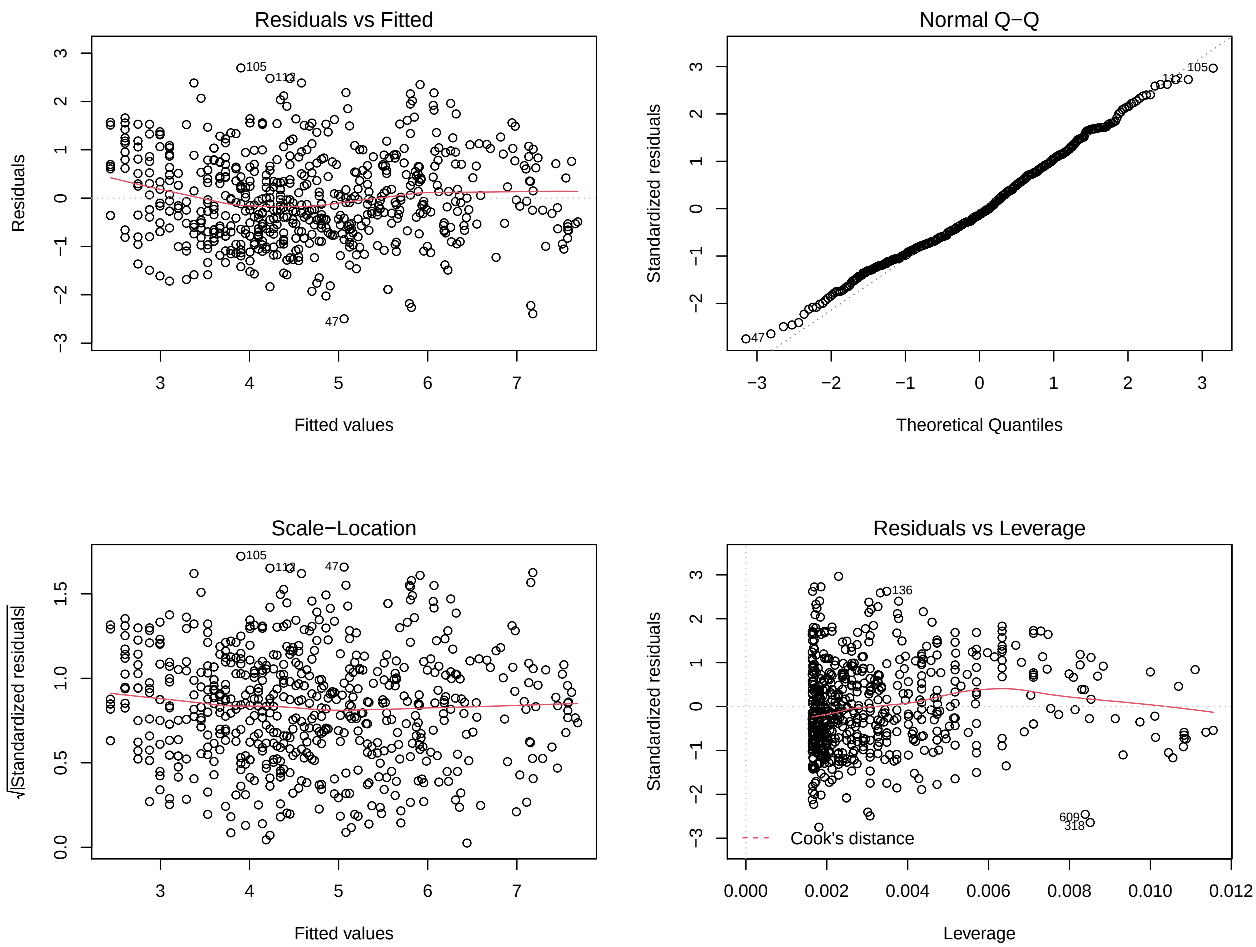
```r
                                          which(data_LOD_subset$log.TropI ==max(data_LOD_subset$log.TropI
row.names(data_LOD_subset2) <- NULL
```

We now re-fit the linear regression model on the reduced dataset and repeat the residual analysis.

```r
trop.fit <- lm( log.TropI ~ log.TropT, data = data_LOD_subset2)
summary(trop.fit)
```

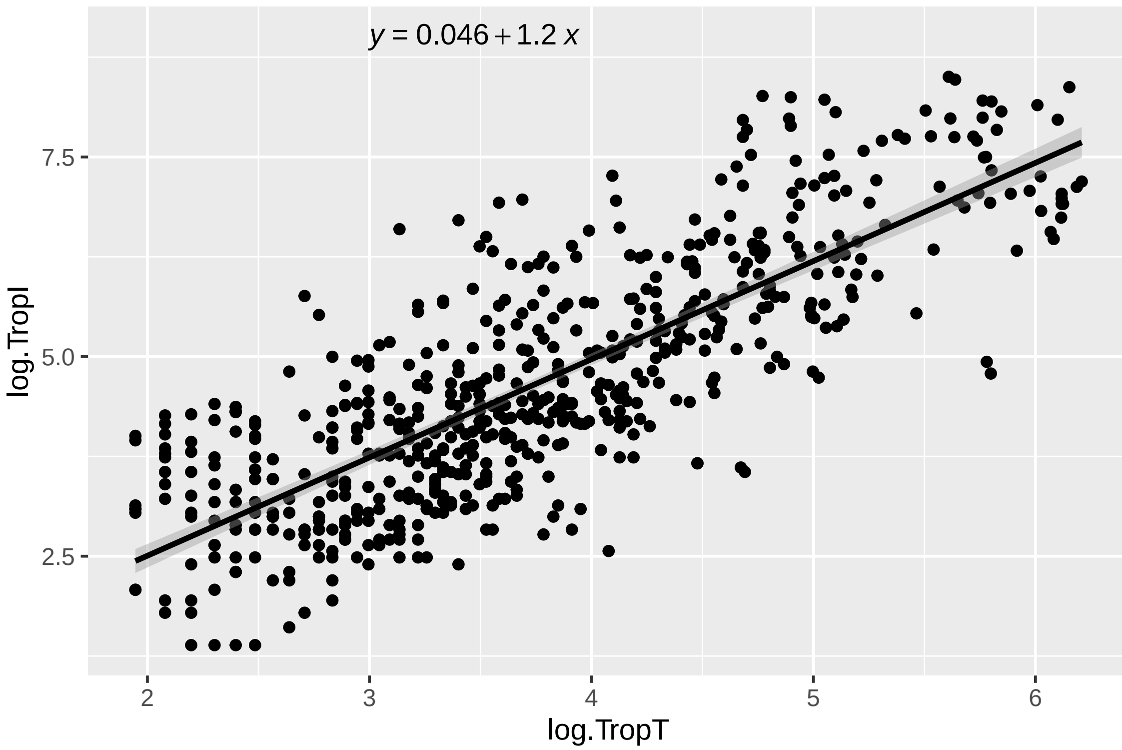
```
##
## Call:
## lm(formula = log.TropI ~ log.TropT, data = data_LOD_subset2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.4966 -0.6568 -0.1129  0.6509  2.6930
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.04595    0.14398   0.319     0.75
## log.TropT    1.23005    0.03699  33.250   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9087 on 608 degrees of freedom
## Multiple R-squared:  0.6452, Adjusted R-squared:  0.6446
## F-statistic:  1106 on 1 and 608 DF,  p-value: < 2.2e-16
```

```r
par(mfrow=c(2,2))
plot(trop.fit)
```



Finally, we plot the data along with the fitted regression line, and estimated regression equation.

```
library(ggplot2)
library(ggpubr)

ggplot(data_LOD_subset2, aes(x = log.TropT, y = log.TropI))+ geom_point() +
  geom_smooth(method="lm", col="black") +
  stat_regline_equation(label.x = 3, label.y = 9)
```



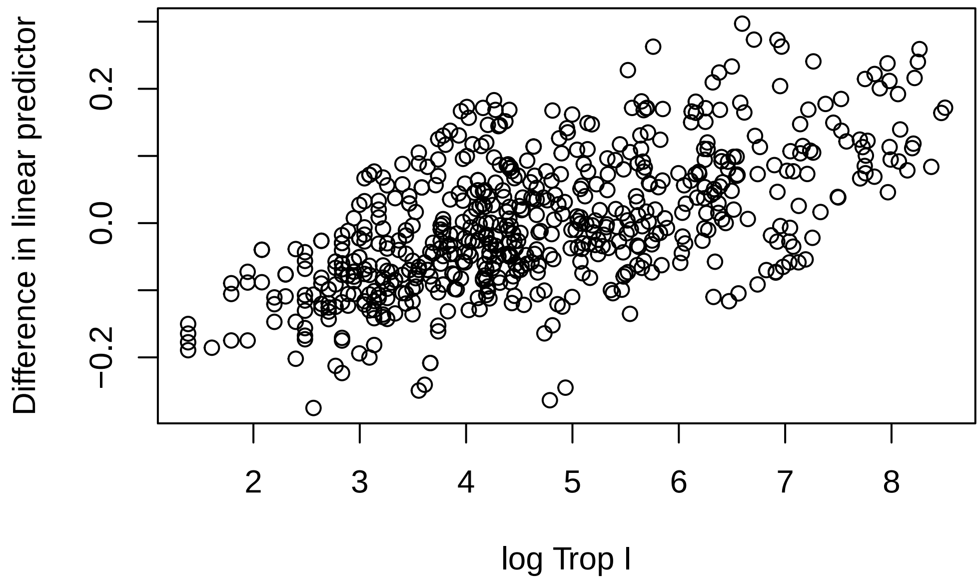
Analysis of predicted values

We compute the predicted log troponin I in our dataset. From this prediction, we can further calculate what will be the difference in the linear predictor of our risk regression model from using the predicted values rather that the observed ones.

```
# Compute predicted log troponin I:
pred_logI <- predict.lm(trop.fit, data_LOD_subset2)

# Difference in linear predictor
dif_LP <- 0.11030707*(data_LOD_subset2$log.TropI - pred_logI)
```

We produce a scatter plot of the observed log troponin I vs the differences computed above.

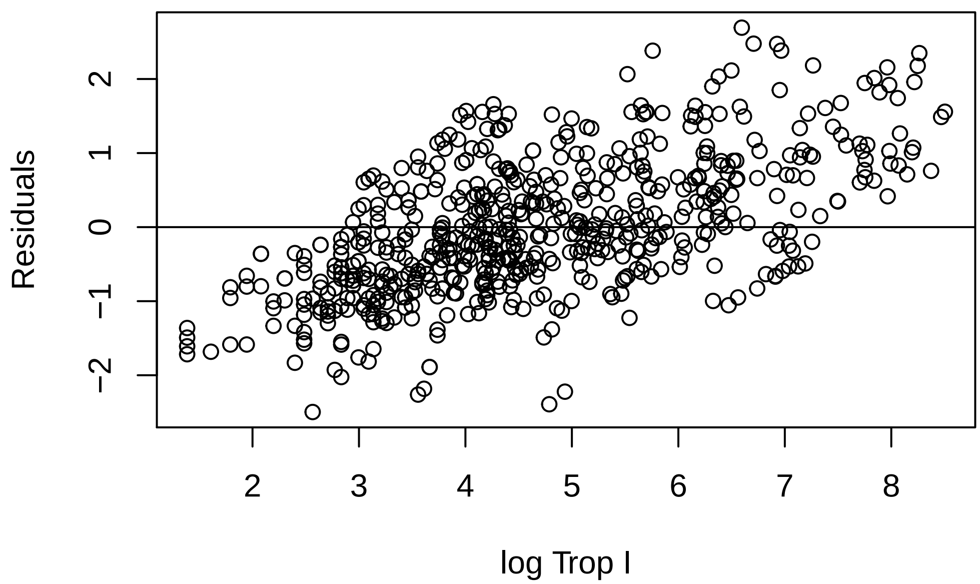
```
df_trops <- cbind(dif_LP, log.tropI=data_LOD_subset2$log.TropI, pred_logI)
df_trops_ordered <- df_trops[order(-df_trops[,1]),]

plot(data_LOD_subset2$log.TropI, dif_LP,
     xlab="log Trop I", ylab = "Difference in linear predictor" )
```



In the above plot, we see that the largest positive differences in the linear predictor of the risk score will be for large values in troponin I.

Finally, we analyse the residuals with respect to the observed troponin I. From the plot below, we expect to underestimate the risk for subjects with small values of troponin I, as a result of the negative differences in the residuals. In addition, we expect an overestimation of the risk for bigger values in log trop I.

```
residuals = data_LOD_subset2$log.TropI-pred_logI

plot(data_LOD_subset2$log.TropI, residuals,
     xlab="log Trop I", ylab="Residuals")
abline(h = 0)
```



Session info

```
## R version 4.0.5 (2021-03-31)
```

```
## Platform: x86_64-apple-darwin17.0 (64-bit)
## Running under: macOS Catalina 10.15.6
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/4.0/Resources/lib/libRblas.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/4.0/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] ggpubr_0.4.0  ggplot2_3.3.5 readr_2.1.2
##
## loaded via a namespace (and not attached):
##  [1] tidyselect_1.1.2 xfun_0.30        purrr_0.3.4      lattice_0.20-45
##  [5] splines_4.0.5    carData_3.0-5    colorspace_2.0-3 vctrs_0.3.8
##  [9] generics_0.1.2   htmltools_0.5.2  yaml_2.3.5       mgcv_1.8-39
## [13] utf8_1.2.2       rlang_1.0.2      pillar_1.7.0     glue_1.6.2
## [17] withr_2.5.0      DBI_1.1.2        bit64_4.0.5      lifecycle_1.0.1
## [21] stringr_1.4.0    munsell_0.5.0    ggsignif_0.6.3   gtable_0.3.0
## [25] evaluate_0.15    labeling_0.4.2   knitr_1.37       tzdb_0.2.0
## [29] fastmap_1.1.0    parallel_4.0.5   fansi_1.0.2      highr_0.9
## [33] broom_0.7.12     polynom_1.4-0    scales_1.1.1     backports_1.4.1
## [37] vroom_1.5.7      abind_1.4-5      farver_2.1.0     bit_4.0.4
## [41] hms_1.1.1        digest_0.6.29    stringi_1.7.6    rstatix_0.7.0
## [45] dplyr_1.0.8      grid_4.0.5       cli_3.2.0        tools_4.0.5
## [49] magrittr_2.0.2   tibble_3.1.6     crayon_1.5.0     tidyr_1.2.0
## [53] car_3.0-12       pkgconfig_2.0.3  Matrix_1.4-0     ellipsis_0.3.2
## [57] assertthat_0.2.1 rmarkdown_2.13   rstudioapi_0.13  R6_2.5.1
## [61] nlme_3.1-155     compiler_4.0.5
```

"ARCHITECT Stat High Sensitivity Troponin-I." https://www.accessdata.fda.gov/cdrh_docs/pdf19/K191595.pdfl.