

Characterising effect of anaemia on mortality in severe malaria

Contents

Background	1
Exploratory analysis	2
Predictive value of anaemia on death adjusting for confounders	2
More complex GAM model	5

Background

This looks at the severe malaria legacy dataset from MORU

The contributions of the different studies:

```
# Whole dataset
table(Leg_data$studyID)

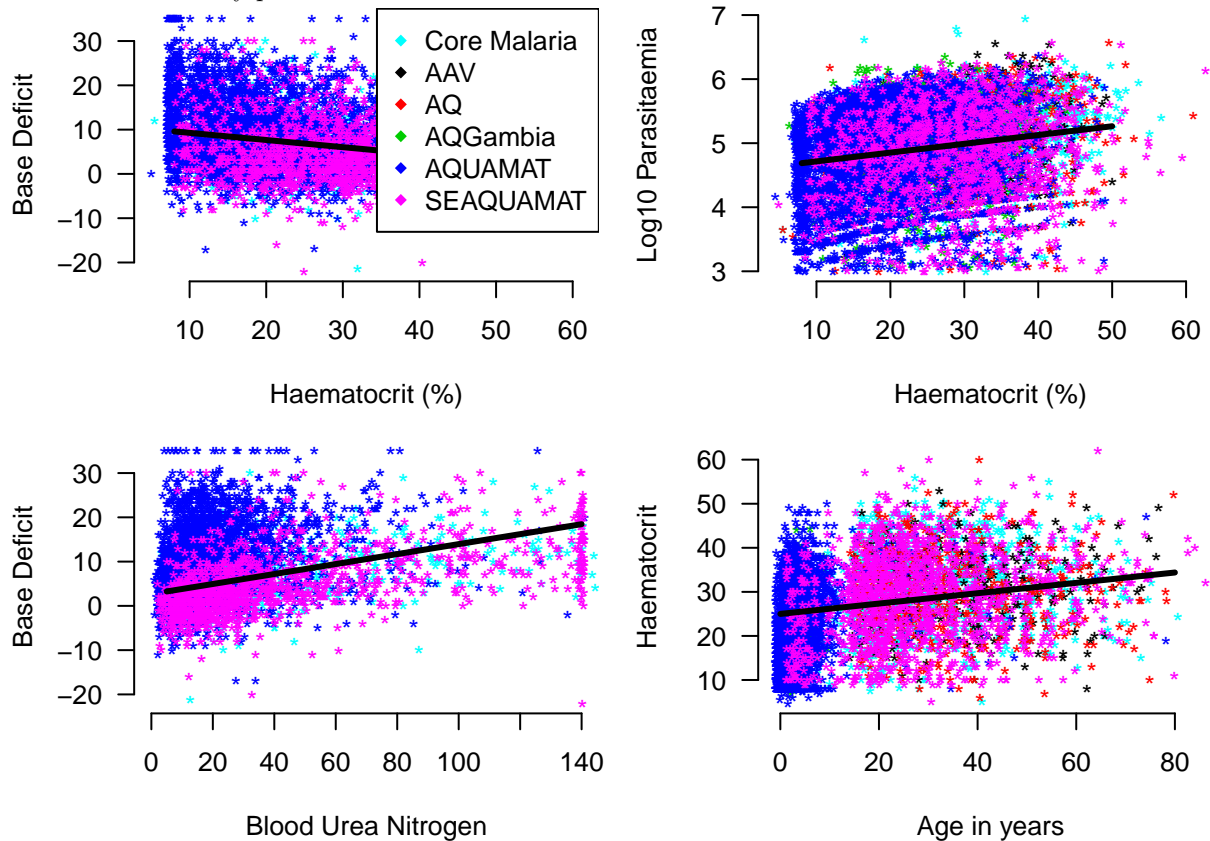
##
##      AAV      AQ    AQGambia    AQUAMAT Core Malaria
##      370      560      579      5494      1121
##    SEQUAMAT
##      1461

# in the complete dataset (all variables recorded)
table(Complete_Leg_data$studyID)

##
##    AQUAMAT Core Malaria    SEQUAMAT
##      3779      359      1090
```

Exploratory analysis

Let's look at the key predictive variables. We use a random effects term to model differences between studies.



Predictive value of anaemia on death adjusting for confounders

Before fitting the more complex GAM models we explore the standard glm (logistic regression) models.

```
mod_full = glmer(outcome ~ HCT + LPAR + AgeInYear + BUN + BD + drug + (1 | studyID),
  data=Complete_Leg_data, family=binomial)
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control
## $checkConv, : unable to evaluate scaled gradient

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control
## $checkConv, : Model failed to converge: degenerate Hessian with 1 negative
## eigenvalues
```

```
summary(mod_full)
```

```
## Warning in vcov.merMod(object, use.hessian = use.hessian): variance-covariance matrix computed from
## not positive definite or contains NA values: falling back to var-cov estimated from RX

## Warning in vcov.merMod(object, correlation = correlation, sigma = sig): variance-covariance matrix co
## not positive definite or contains NA values: falling back to var-cov estimated from RX

## Generalized linear mixed model fit by maximum likelihood (Laplace
## Approximation) [glmerMod]
```

```

## Family: binomial ( logit )
## Formula:
## outcome ~ HCT + LPAR + AgeInYear + BUN + BD + drug + (1 | studyID)
## Data: Complete_Leg_data
##
##      AIC      BIC    logLik deviance df.resid
##  3199.6   3278.3 -1587.8   3175.6     5216
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.7825 -0.3409 -0.2313 -0.1633 10.1058
##
## Random effects:
## Groups Name          Variance Std.Dev.
## studyID (Intercept) 0.07967  0.2823
## Number of obs: 5228, groups: studyID, 3
##
## Fixed effects:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -1.850e+01  5.664e+02  -0.033  0.97394
## HCT            1.960e-02  5.471e-03   3.583  0.00034 ***
## LPAR           1.855e-02  6.719e-02   0.276  0.78247
## AgeInYear      2.186e-02  4.343e-03   5.033  4.83e-07 ***
## BUN            1.166e-02  1.694e-03   6.882  5.89e-12 ***
## BD             1.361e-01  6.944e-03  19.601 < 2e-16 ***
## drugArtesunate 1.404e+01  5.664e+02   0.025  0.98022
## drugChloroquine 1.610e+01  5.664e+02   0.028  0.97733
## drugLumefantrine -1.034e+00  4.187e+03   0.000  0.99980
## drugNAC        -5.245e+00  8.479e+03  -0.001  0.99951
## drugQuinine     1.438e+01  5.664e+02   0.025  0.97975
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##      (Intr) HCT    LPAR   AgInYr BUN    BD      drgArt drgChl drgLmf
## HCT      0.000
## LPAR     -0.001 -0.172
## AgeInYear 0.000 -0.183  0.030
## BUN       0.000  0.064 -0.050 -0.189
## BD        0.000  0.263 -0.135  0.120 -0.263
## drugArtesnt -1.000  0.000  0.000  0.000  0.000  0.000
## drugChlorqn -1.000  0.000  0.000  0.000  0.000  0.000  1.000
## drugLmfntn -0.135  0.000  0.000  0.000  0.000  0.000  0.135  0.135
## drugNAC    -0.067  0.000  0.000  0.000  0.000  0.000  0.067  0.067  0.009
## drugQuinine -1.000  0.000  0.000  0.000  0.000  0.000  1.000  1.000  0.135
##      drgNAC
## HCT
## LPAR
## AgeInYear
## BUN
## BD
## drugArtesnt
## drugChlorqn
## drugLmfntn

```

```
## drugNAC
## drugQuinine 0.067
## convergence code: 0
## unable to evaluate scaled gradient
## Model failed to converge: degenerate Hessian with 1 negative eigenvalues
```

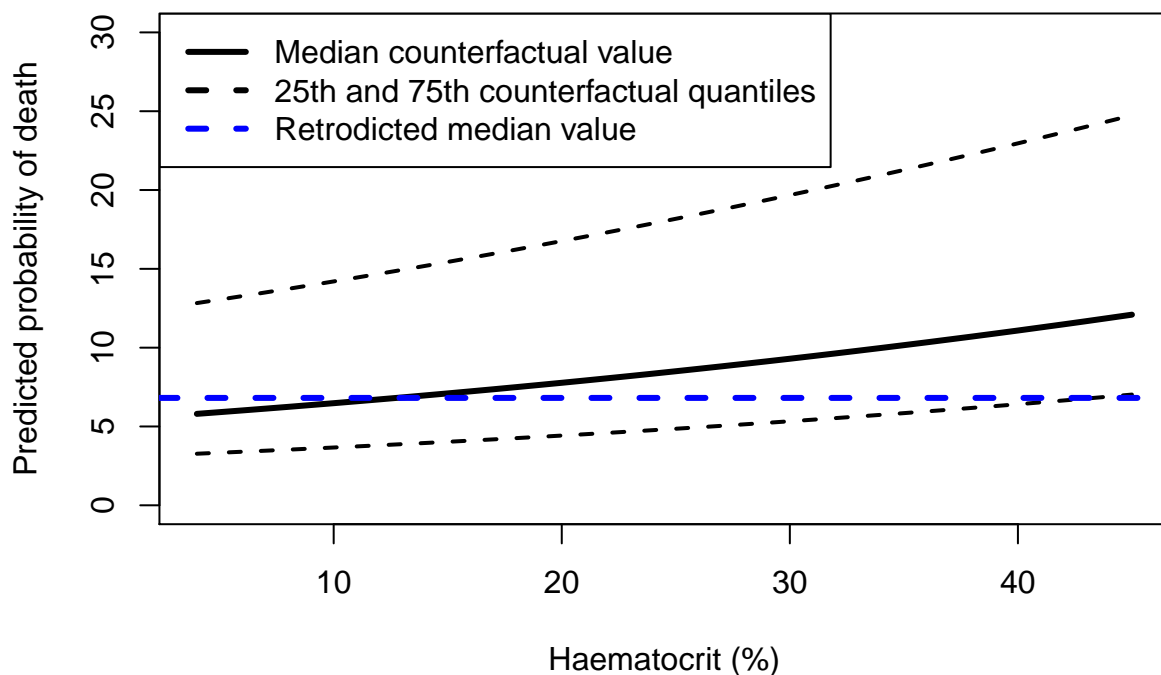
Now let's make counterfactual predictions of anaemia on death for the patients in the database.

```
myquantiles = c(0.25,0.5,0.75) # this is 50% predictive interval

overall_median_mortality = median(100*predict(mod_full, type='response'))
par(las=1, bty='n')
x_hcts = seq(4,45, by=1)
probs_lin = array(dim = c(3, length(x_hcts)))
for(i in 1:length(x_hcts)){
  mydata = Complete_Leg_data
  mydata$HCT=x_hcts[i]
  ys = 100*predict(mod_full, newdata = mydata, re.form=NA, type='response')
  probs_lin[,i] = quantile(ys, probs=myquantiles)
}
```

The way to interpret this 'counterfactual' plot is as follows: suppose that every individual in the dataset was assigned (as in a intervention) a specific haematocrit X , what would the resulting per patient probability of death be. Here we summarise these probabilities by the predicted mean probability of death and 80% predictive intervals.

```
plot(x_hcts,probs_lin[2,], xlim=c(4,45), ylab='Predicted probability of death',
     xlab='Haematocrit (%)', ylim=c(0,30), lty=1, lwd=3, type='l')
lines(x_hcts, probs_lin[1,], lty=2, lwd=2)
lines(x_hcts, probs_lin[3,], lty=2, lwd=2)
abline(h=overall_median_mortality, lwd=3, col='blue',lty=2)
legend('topleft', col=c('black','black','blue'), lwd=3, lty=c(1,2,2),
      legend = c('Median counterfactual value', '25th and 75th counterfactual quantiles', 'Retrodicted median value'))
```



More complex GAM model

The GAM model allows for non-linear relationships between certain variables and the outcome.

Here we fit as non-linear the effect of age and haematocrit on mortality.

```
mod_full_GAM = gam(outcome ~ s(HCT, AgeInYear) + LPAR + BUN + BD,
                    data=Complete_Leg_data, family=binomial)
summary(mod_full_GAM)
```

```
##
## Family: binomial
## Link function: logit
##
## Formula:
## outcome ~ s(HCT, AgeInYear) + LPAR + BUN + BD
##
## Parametric coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.706458   0.332187 -11.158  < 2e-16 ***
## LPAR         -0.003305   0.067265  -0.049    0.961
## BUN          0.010038   0.001720   5.838 5.29e-09 ***
## BD           0.140822   0.007133  19.742  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##              edf Ref.df Chi.sq p-value
## s(HCT, AgeInYear) 6.522   9.09  131.3  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.202   Deviance explained = 21.4%
## UBRE = -0.38846   Scale est. = 1          n = 5228
```

Now we compute the corresponding counterfactual probabilities of death for the dataset for all values of the haematocrit:

```
overall_median_mortalityGAM = median(100*predict(mod_full_GAM, type='response'))
par(las=1, bty='n')
probs_gam = array(dim = c(3, length(x_hcts)))
for(i in 1:length(x_hcts)){
  mydata = Complete_Leg_data
  mydata$HCT=x_hcts[i]
  ys = 100*predict(mod_full_GAM, newdata = mydata, re.form=NA, type='response')
  probs_gam[,i] = quantile(ys, probs=myquantiles)
}
```

We see that the effect of haematocrit on mortality is non-linear under this model: below 20 is protective, above 20 plateaus out:

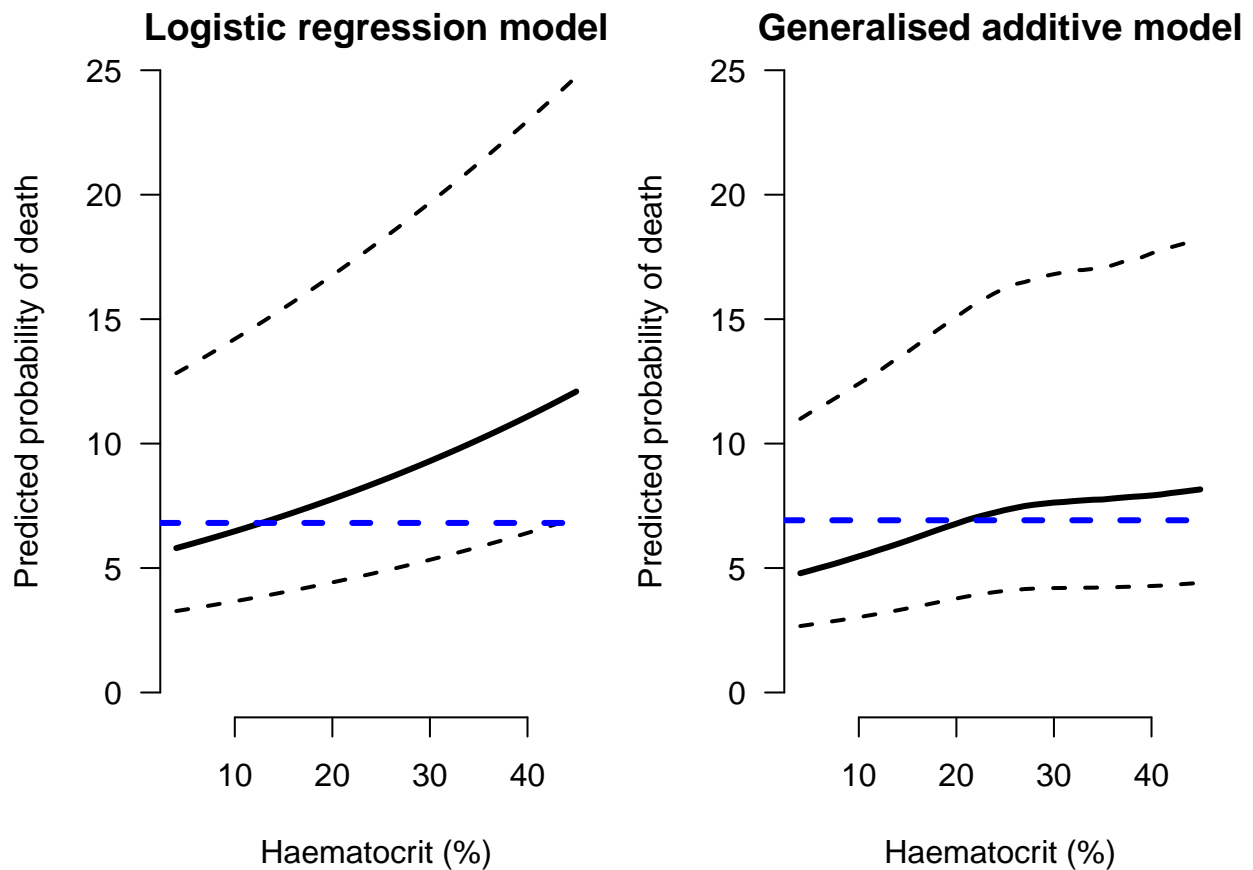
```
#
par(las=1, mfrow=c(1,2), bty='n', mar=c(4,4,1,1))

plot(x_hcts, probs_lin[2,], xlim=c(4,45), ylab='Predicted probability of death',
     xlab='Haematocrit (%)', ylim=c(0,25), lty=1, lwd=3, type='l')
```

```

lines(x_hcts, probs_lin[1,], lty=2, lwd=2)
lines(x_hcts, probs_lin[3,], lty=2, lwd=2)
abline(h=overall_median_mortality, lwd=3, col='blue', lty=2)
title('Logistic regression model')
### And now the GAM model
plot(x_hcts, probs_gam[2,], xlim=c(4,45), ylab='Predicted probability of death',
     xlab='Haematocrit (%)', ylim=c(0,25), lty=1, lwd=3, type='l')
lines(x_hcts, probs_gam[1,], lty=2, lwd=2)
lines(x_hcts, probs_gam[3,], lty=2, lwd=2)
abline(h=overall_median_mortalityGAM, lwd=3, col='blue', lty=2)
title('Generalised additive model')

```



```

#legend('bottomright', col=c('black','black','blue'), lwd=3, lty=c(1,2,2),
#      legend = c('Mean predicted mortality', '80% predicted interval','Observed #mortality'))

```