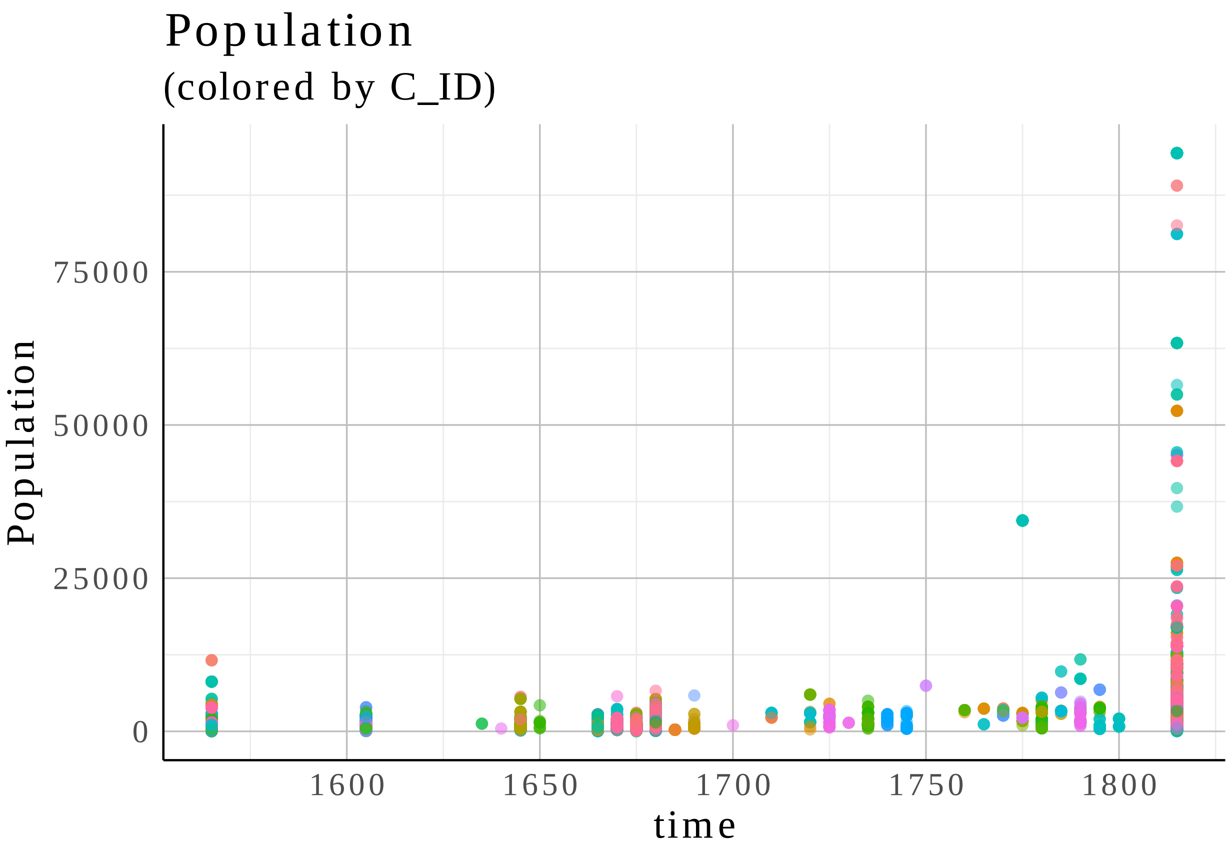
results

Sven Maurice Morlock

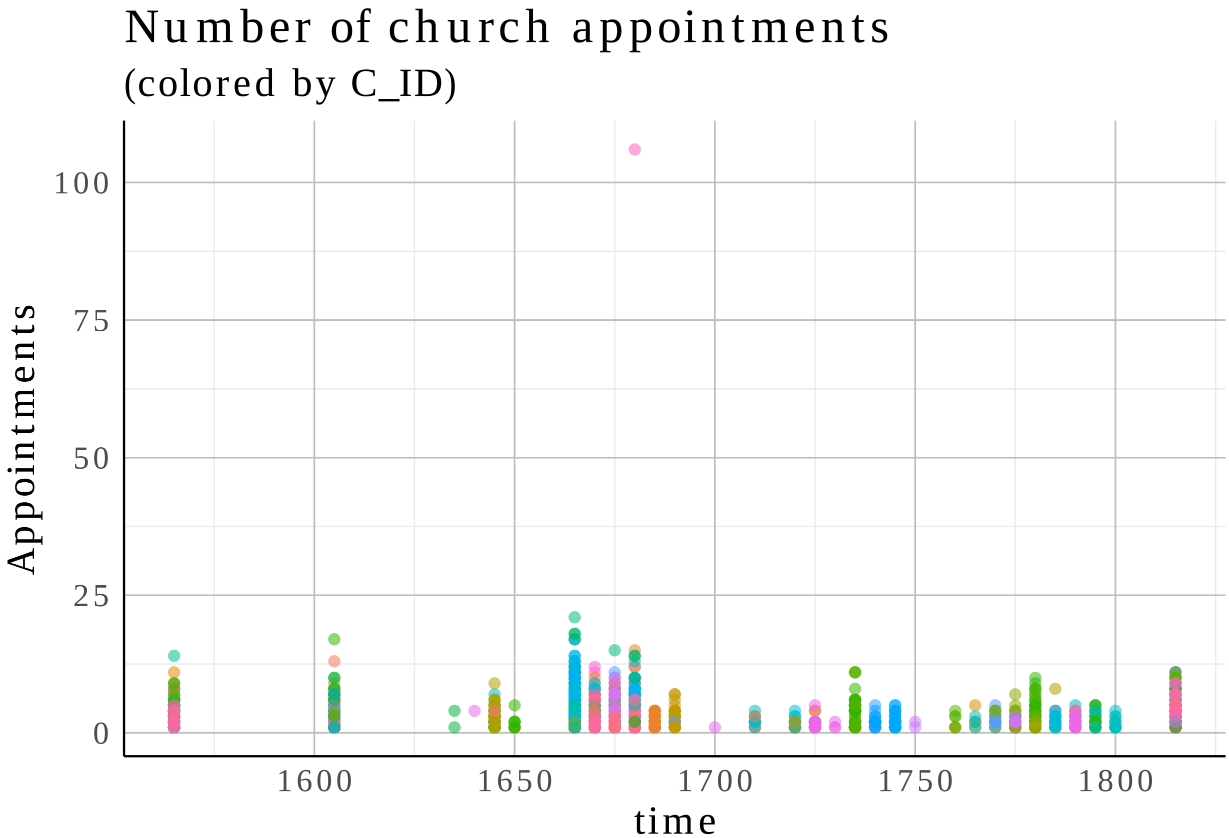
2022-08-26

```
## Data: data
## Models:
## model.ols.1: Population ~ Appointments
## model.linear.1: Population ~ Appointments + (1 | C_ID)
##                npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## model.ols.1       3 286402 286425 -143198   286396
## model.linear.1    4 279181 279211 -139586   279173 7223.4  1  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The following text contains information on the results contained in the script `main.R`. Specifically the relation between church appointments and population in England are analyzed here. The following graphic illustrates the population development in England during the time period of 1525 and 1850. The colors correspond to the CCE_Id, a unique identifier of a church parish in England:

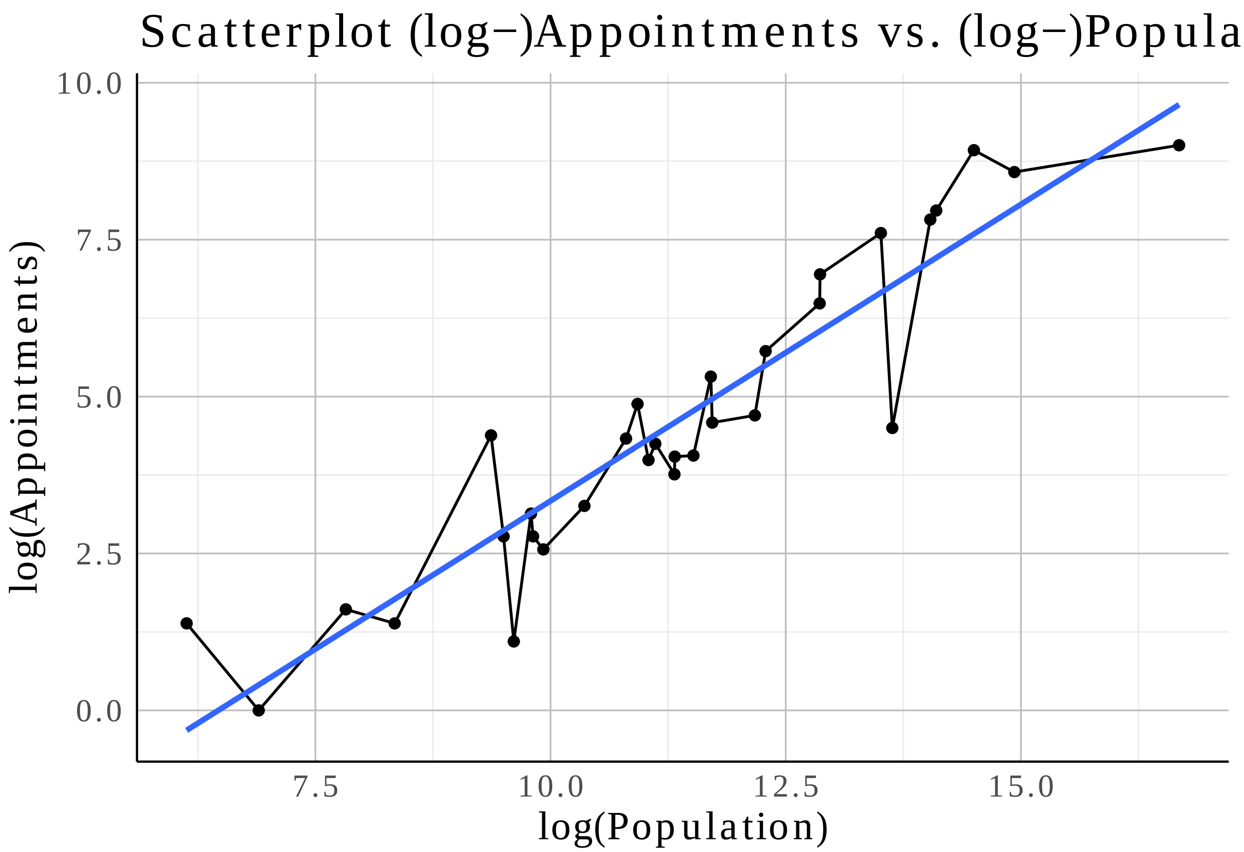


As one can see, especially in the late periods population in England was growing. The following graphic shows the development of church appointments during this time span, again colored by the identifier of church parish:



Since the number of church appointments does not show the same development as population does we can look at a scatter plot, containing the logarithmic number of appointments on the y-axis and the logarithmic population number on th x-axis:

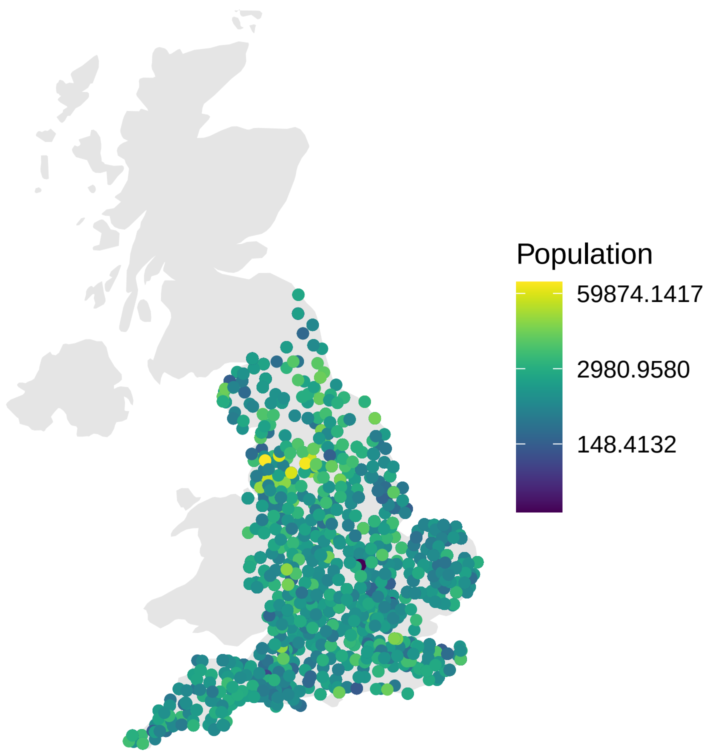
```
## `geom_smooth()` using formula 'y ~ x'
```



The linear regression line indicates a positive association between population and church appointments. The following graphic illustrates the development of population in different regions in England. The depicts the population size:

```
## Warning: Removed 3 rows containing missing values (geom_point).
```

Variation in Population



In the following the parish is taken into account as a fixed effect using the `nlme4` package: the variable `model.ols.1` is a simple linear regression of Population on Appointments. The variable `model.linear.1` contains a fitted linear model but taking with fixes effects.

The simple ols regression yields the following result:

```
##
## Call:
## lm(formula = Population ~ Appointments, data = data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -2064  -1509  -1099   -253  92567
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2169.40      77.48  27.998   <2e-16 ***
## Appointments   -59.99      26.41  -2.272   0.0231 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5997 on 14151 degrees of freedom
## Multiple R-squared:  0.0003646,  Adjusted R-squared:  0.000294
## F-statistic: 5.162 on 1 and 14151 DF,  p-value: 0.0231
```

The fixed effect regression yields:

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: Population ~ Appointments + (1 | C_ID)
##    Data: data
```

```
##
##       AIC       BIC    logLik  deviance  df.resid
##  279180.9  279211.2 -139586.5  279172.9     14149
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -10.0662  -0.1138  -0.0400   0.0260  16.0243
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  C_ID     (Intercept) 12728655 3568
##  Residual             19147866 4376
## Number of obs: 14153, groups:  C_ID, 858
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)   2335.92     143.95  16.227
## Appointments   -17.53      20.09  -0.873
##
## Correlation of Fixed Effects:
##             (Intr)
## Appointmnts -0.308
```

The percentage of total variation in the data that can be attributed to the fixed effects is therefore 0.3993113. So roughly 40 percent. Since the simple ols is nested within the fixed effect regression an (approximate) likelihood ratio test can be conducted:

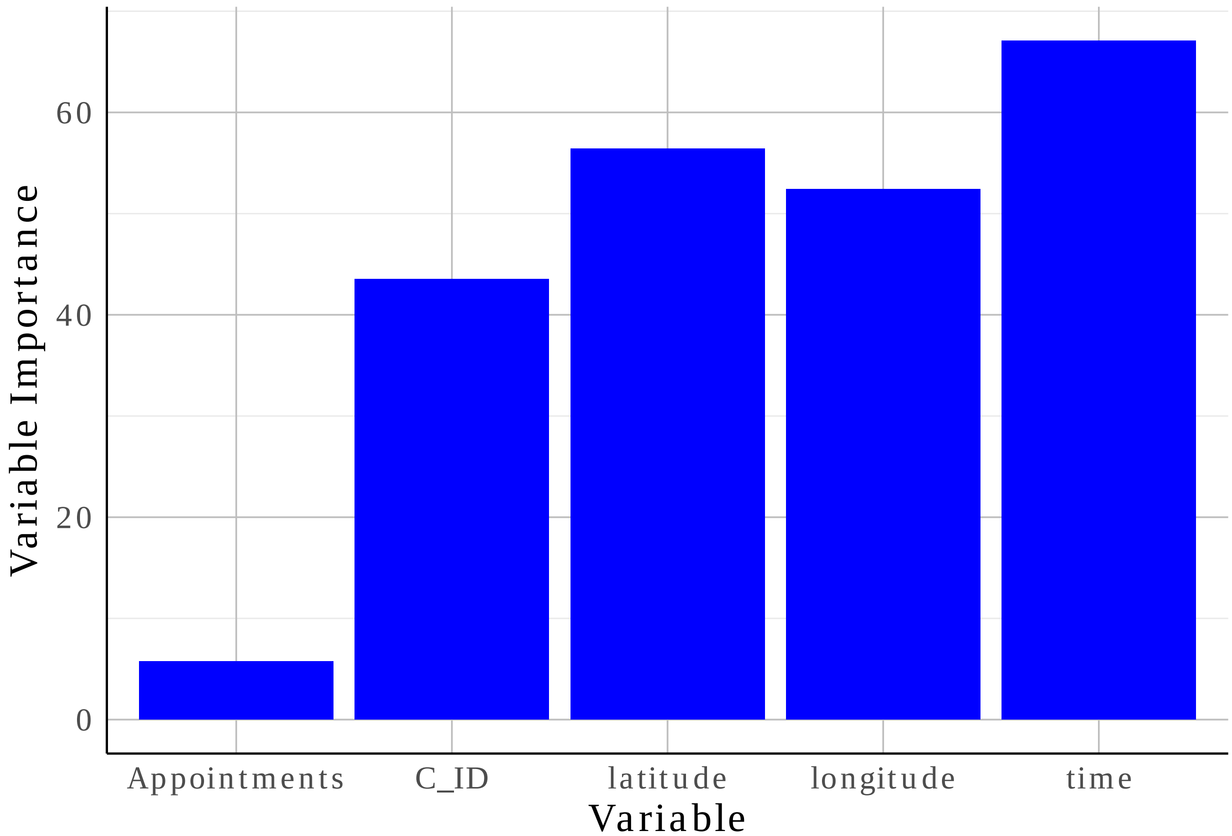
```r
anova(model.linear.1, model.ols.1)
```

```
## Data: data
## Models:
## model.ols.1: Population ~ Appointments
## model.linear.1: Population ~ Appointments + (1 | C_ID)
##                npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## model.ols.1       3 286402 286425 -143198   286396
## model.linear.1    4 279181 279211 -139586   279173 7223.4  1  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The corresponding p-value is smaller than any typical significance level, indicating that the fixed effects contribute to a significant extent to the overall data variation given the number of appointments.

Therefore a simple linear regression is used in the following. One can ask how well number of church appointments do predict the population in out data. Therefore the data is split into a test and training set. Threee models are considered: - a simple linear model, model.linear - a random forest model, model.rforest - a Extreme Gradient Boosting model, model.xgb

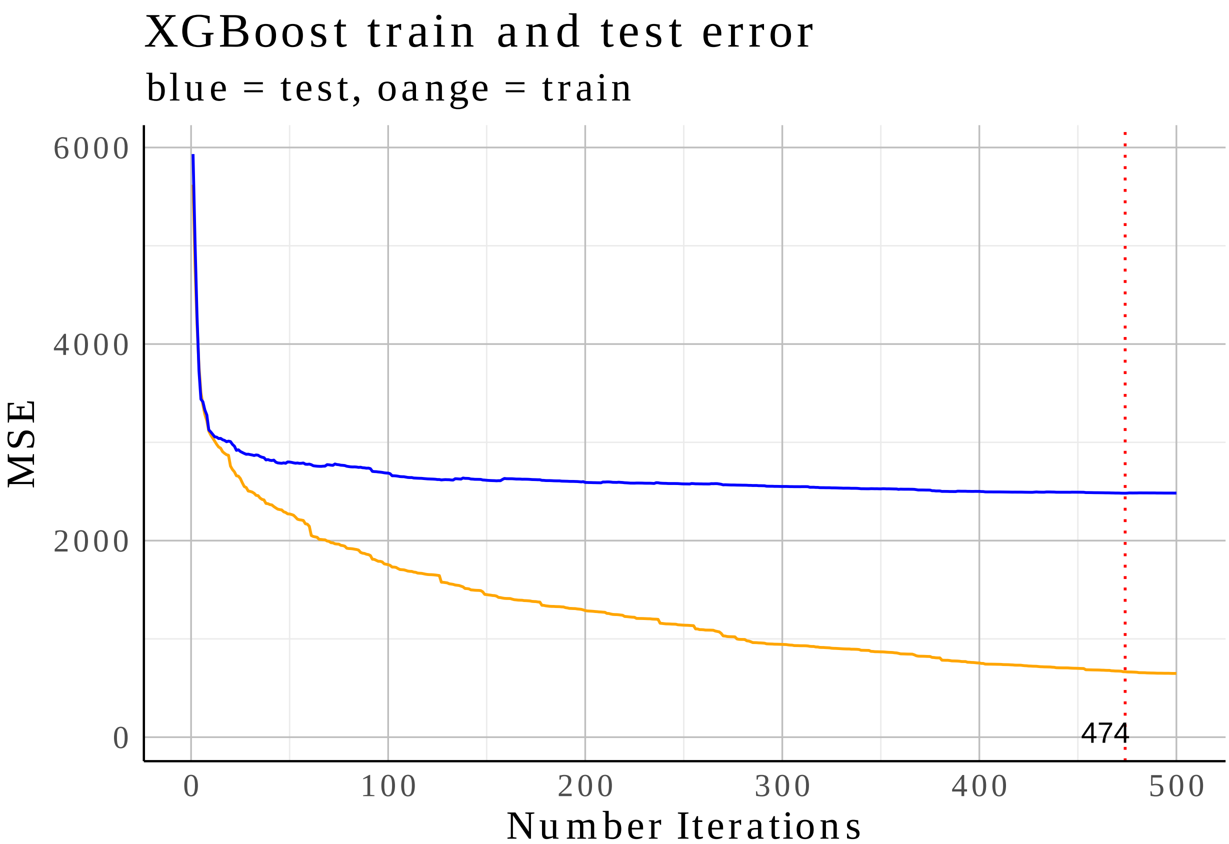
The following plot shows the feature importance of the variables used in random forest:




```r
xgb.train <- xgb.DMatrix(data = data.matrix(X.train), label = y.train)
xgb.test <- xgb.DMatrix(data = data.matrix(X.test), label = y.test)
model.xgb <- xgb.train(data = xgb.train, max.depth = 3, nrounds = 500, watchlist = list(train=xgb.train
```

As one can see, taken into account the other variables the number of appointments does contribute relatively less to the prediction of population.

Extreme Gradient Boosting is an Ensemble method that successively trains on weak base learning by adapting the weight to variables that contribute most the the reduction in resulting estimation error. Here regression tree of depth three was chosen. To determine the number of boosting iterations consider the following graphic:



Using 300 hundred iterations the test error is smallest at 298 iterations which is used as hyper parameter in the final xgboost model.

The following table shows the results on test data for the three models

```
##     Error Linear Model Random Forest         XGB
## 1:   MSE 124017183875   35147797751 21340518784
## 2:   MAE      6648083       3113312     1990910
```