

# Base model fitting for RTC Data

220209395

2023-03-19

## Introduction

This report details the baseline process for building a GLM model which is to be used in the MAS61004 group project.

## Data Cleaning

We begin by converting to a binary variable by severity. Accidents with a severity of 3 become a 0, and others are converted to a 1. This is done with a simple ifelse statement, adding a new column to the accidents data.

```
accidents <- accidents %>%
  mutate(bin_severity = ifelse(.data$accident_severity == 3, 0, 1))
```

We now reduce the size of the dataset to contain only variables we care about, as agreed upon and described in section 3 of the final report. The “sub\_X” objects are then combined into one large dataset. Note that on creation of the main dataset (called df) we remove any sex of casualty variables that take on a value of 9 as these are uninformative and so can be excluded.

```
sub_accidents <- accidents %>%
  select(
    "accident_index",
    "bin_severity",
    "day_of_week",
    "road_type",
    "speed_limit",
    "time",
    "light_conditions",
    "weather_conditions",
    "local_authority_ons_district",
    "road_surface_conditions"
  )

sub_vehicles <- vehicles %>%
  select(
    "accident_index",
    "sex_of_driver",
    "age_band_of_driver"
  )

sub_casualties <- casualty %>%
  select(
    "accident_index",
    "sex_of_casualty",
```

```

    "age_band_of_casualty"
)

df_sub <- merge.data.frame(sub_vehicles, sub_casualties, by = "accident_index")
df_sub_unique <- unique(df_sub)
df <- merge.data.frame(sub_accidents, df_sub_unique, by = "accident_index") %>%
  filter(sex_of_casualty != 9)

```

Before dealing with other variables in the dataset, we deal with time on its own. The times provided in the dataset are characters, for example of the form “12:14”. There are many unique times in the dataset, which would cause issues when fitting models as these would all be treated as individual factors. We therefore split the 24 hour day into 6 chunks of 4 hours, as this dramatically reduces the number of factors, but also gives good information about the relationship between time of day and accident severity.

The first thing to do is build a function to parse the times by splitting the character before the colon, and converting that to an integer. A series of if-statements are then checked to put the time into the correct category. This isn’t the most computationally efficient method, but it does the job. We then apply this function to every time observation to create a new column of data called “time\_group”, and drop the original time variable from the data so it doesn’t cause problems later on.

```

# Split time into 6 chunks of four hours
parse_times <- function(time){
  split_time <- stringr::str_split(time, ":", simplify = TRUE)
  hour <- as.integer(split_time[1])
  if (hour < 4) return(1)
  if (hour < 8 ) return(2)
  if (hour < 12) return(3)
  if (hour < 16) return(4)
  ifelse(hour < 20, return(5), return(6))
}

df <- df %>%
  mutate(time_group = purrr::map_int(time, parse_times)) %>%
  select(-time)

```

Location is to be included, but in the current state, the *local\_authority\_ons\_district* variable has too many unique values. To account for this, the region codes (those beginning with E0, N, S or W) are matched to a wider local area, such as the West Midlands, Yorkshire and The Humber etc. Across England, Wales, Scotland & Northern Ireland, there are 12 unique regions. Each region is assigned a numerical valueable. This assignment is done alphabetically, so East Midlands is 1, all the way through to Yorkshire and The Humber at 12. With this in mind, we can add region code to the data. The original location data is then dropped from df.

```

# Obtains the numerical value of a region from a given DNS coded
obtain_region <- function(region_code){
  return(regions$Region.Code[which(regions$Code == region_code)])
}

df <- df %>%
  mutate(region = purrr::map_int(local_authority_ons_district, obtain_region)) %>%
  select(-local_authority_ons_district)

```

All variables except speed limit need to be factors. To achieve this we mutate across all integer columns and change the type to factor. However, er need speed limit to stay as an integer to make the model adaptable to non-standard speed limits. The “bin\_severity” column needs to be a factor, so we do this by calling the as.factor function once more just to ensure it is infact a factor. By viewing the head of the data as a tibble

we can check that the columns are of expected type.

```
df <- df %>%
  mutate(across(where(is.integer), as.factor))
df$speed_limit <- as.integer(df$speed_limit) #Speed limit goes back to integer
df$bin_severity <- as.factor(df$bin_severity)
head(tibble(df))

## # A tibble: 6 x 14
##   accident_index bin_s~1 day_o~2 road_~3 speed~4 light~5 weath~6 road_~7 sex_o~8
##   <chr>          <fct>    <fct>    <fct>    <int> <fct>    <fct>    <fct>
## 1 2020010219808 0         3         6         2 1     9         9         2
## 2 2020010220496 0         2         6         2 1     1         1         1
## 3 2020010228005 0         4         6         3 4     1         2         3
## 4 2020010228006 1         4         6         3 4     1         1         1
## 5 2020010228011 0         4         6         3 4     1         1         1
## 6 2020010228012 0         4         2         2 4     1         1         1
## # ... with 5 more variables: age_band_of_driver <fct>, sex_of_casualty <fct>,
## #   age_band_of_casualty <fct>, time_group <fct>, region <fct>, and abbreviated
## #   variable names 1: bin_severity, 2: day_of_week, 3: road_type,
## #   4: speed_limit, 5: light_conditions, 6: weather_conditions,
## #   7: road_surface_conditions, 8: sex_of_driver
```

Some observations of the data are recorded as -1. We don't want these in the data, so we trim the data down to remove any rows that contain -1 in any of the columns. This helps with run-time of the model as well due to the size of the data before this is done. At this stage we also remove the accident index variable, as it was only needed for combining the three datasets, and isn't needed in fitting the GLM itself.

```
has.neg <- apply(df, 1, function(row) any(row == -1))
df <- df[-which(has.neg), ] %>%
  select(-accident_index)
```

## Fitting

We fit a logistic regression to the data using the stats::glm function. We specify that the response variable is binomial, and that we wish to use a logit link function. No other interaction terms are included, as this will be built upon later.

```
fit <- glm(
  data = df,
  formula = bin_severity ~ .,
  family = binomial(link = "logit"))
)
```

Let us take a look at the summary and plots of this model.

```
summary(fit)

##
## Call:
## glm(formula = bin_severity ~ ., family = binomial(link = "logit"),
##      data = df)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.5739  -0.7530  -0.6218  -0.4157   2.8225
##
```

```

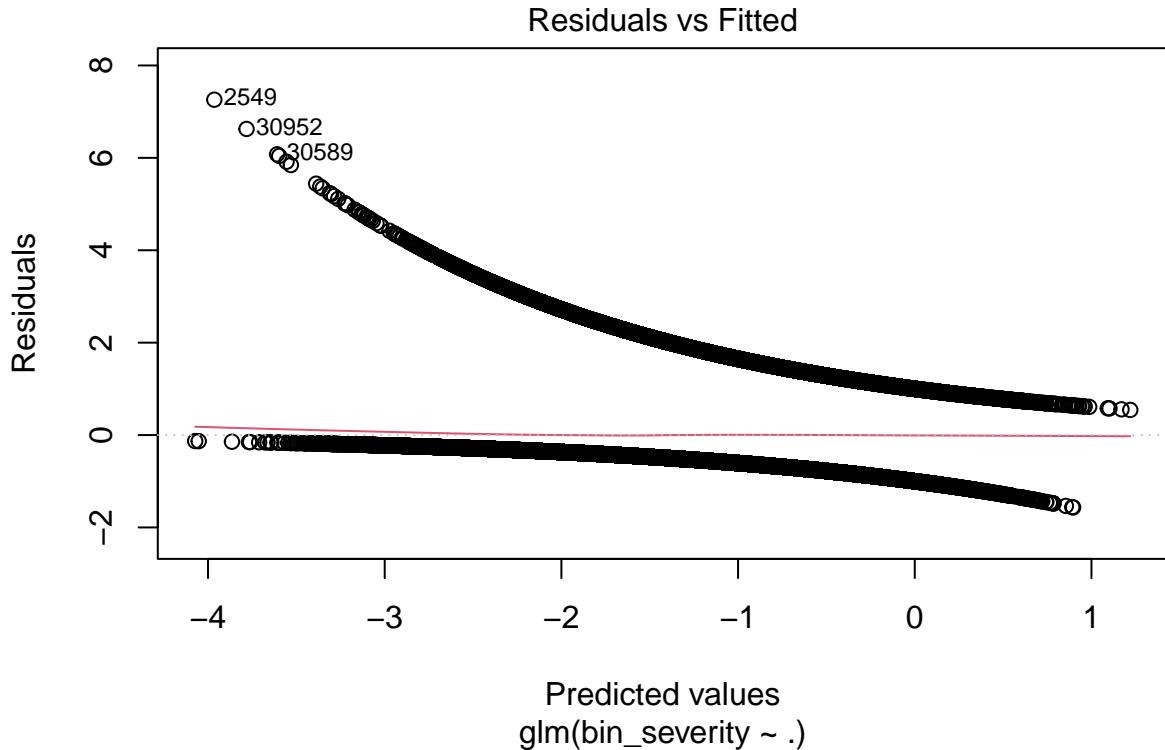
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                 -1.644006   0.431747 -3.808 0.000140 ***
## day_of_week2                  -0.178770   0.023107 -7.737 1.02e-14 ***
## day_of_week3                  -0.187139   0.023000 -8.136 4.07e-16 ***
## day_of_week4                  -0.192339   0.022858 -8.415 < 2e-16 ***
## day_of_week5                  -0.118646   0.022515 -5.270 1.37e-07 ***
## day_of_week6                  -0.141966   0.022166 -6.405 1.51e-10 ***
## day_of_week7                  -0.038302   0.022692 -1.688 0.091439 .
## road_type2                      0.323376   0.060938  5.307 1.12e-07 ***
## road_type3                      0.061049   0.032112  1.901 0.057282 .
## road_type6                      0.488735   0.028699 17.030 < 2e-16 ***
## road_type7                      0.018231   0.055495  0.329 0.742525
## road_type9                      -0.312001   0.084522 -3.691 0.000223 ***
## speed_limit                     0.188456   0.004715 39.972 < 2e-16 ***
## light_conditions4                0.040551   0.019996  2.028 0.042565 *
## light_conditions5                0.053542   0.068794  0.778 0.436401
## light_conditions6                0.240118   0.026899  8.927 < 2e-16 ***
## light_conditions7                -0.473057   0.056234 -8.412 < 2e-16 ***
## weather_conditions2               -0.124128   0.022273 -5.573 2.50e-08 ***
## weather_conditions3               0.093784   0.145102  0.646 0.518062
## weather_conditions4               0.183752   0.044103  4.166 3.09e-05 ***
## weather_conditions5               -0.036287   0.044353 -0.818 0.413276
## weather_conditions6               0.539645   0.209884  2.571 0.010136 *
## weather_conditions7               -0.010957   0.073136 -0.150 0.880910
## weather_conditions8               -0.068316   0.040012 -1.707 0.087748 .
## weather_conditions9               -0.365490   0.055126 -6.630 3.35e-11 ***
## road_surface_conditions2          -0.027202   0.016692 -1.630 0.103172
## road_surface_conditions3          -1.078139   0.195372 -5.518 3.42e-08 ***
## road_surface_conditions4          -0.276089   0.067228 -4.107 4.01e-05 ***
## road_surface_conditions5          -0.220648   0.132876 -1.661 0.096801 .
## road_surface_conditions9          -0.086495   0.137311 -0.630 0.528749
## sex_of_driver2                   -0.166366   0.014417 -11.540 < 2e-16 ***
## sex_of_driver3                   -0.479985   0.060223 -7.970 1.59e-15 ***
## age_band_of_driver2                -0.153648   0.460146 -0.334 0.738446
## age_band_of_driver3                0.107394   0.436205  0.246 0.805527
## age_band_of_driver4                0.149958   0.431535  0.347 0.728216
## age_band_of_driver5                0.161412   0.431280  0.374 0.708208
## age_band_of_driver6                0.144557   0.431023  0.335 0.737337
## age_band_of_driver7                0.141425   0.431086  0.328 0.742862
## age_band_of_driver8                0.201195   0.431147  0.467 0.640750
## age_band_of_driver9                0.282638   0.431266  0.655 0.512230
## age_band_of_driver10               0.268665   0.431707  0.622 0.533724
## age_band_of_driver11               0.172723   0.432274  0.400 0.689475
## sex_of_casualty2                  -0.349442   0.013490 -25.905 < 2e-16 ***
## age_band_of_casualty2              -0.093993   0.064138 -1.465 0.142788
## age_band_of_casualty3              0.082312   0.058835  1.399 0.161807
## age_band_of_casualty4              -0.050879   0.052633 -0.967 0.333709
## age_band_of_casualty5              -0.184877   0.051735 -3.574 0.000352 ***
## age_band_of_casualty6              -0.215375   0.049903 -4.316 1.59e-05 ***
## age_band_of_casualty7              -0.165505   0.050590 -3.271 0.001070 **
## age_band_of_casualty8              -0.082391   0.050767 -1.623 0.104607
## age_band_of_casualty9              0.089703   0.051712  1.735 0.082798 .
## age_band_of_casualty10             0.312644   0.054737  5.712 1.12e-08 ***

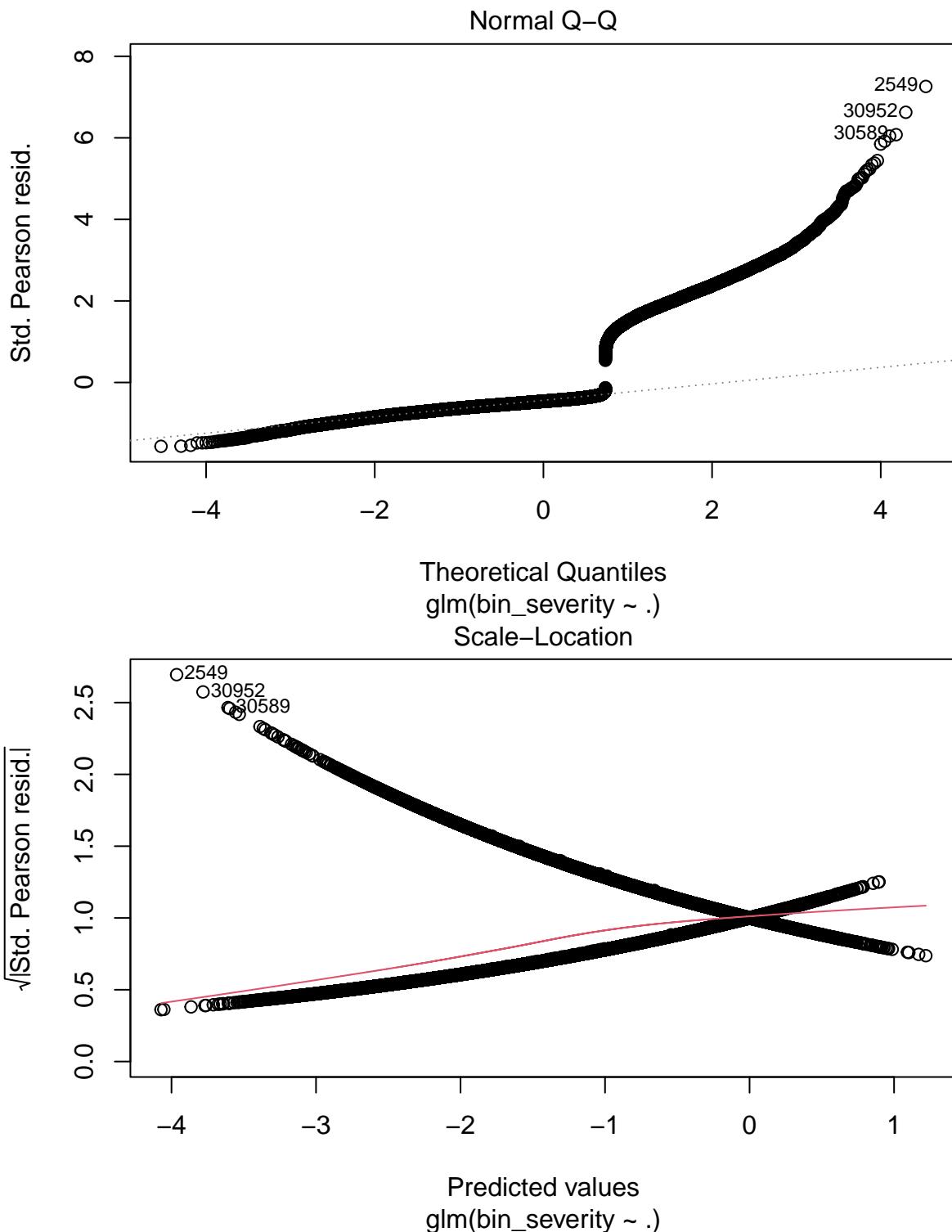
```

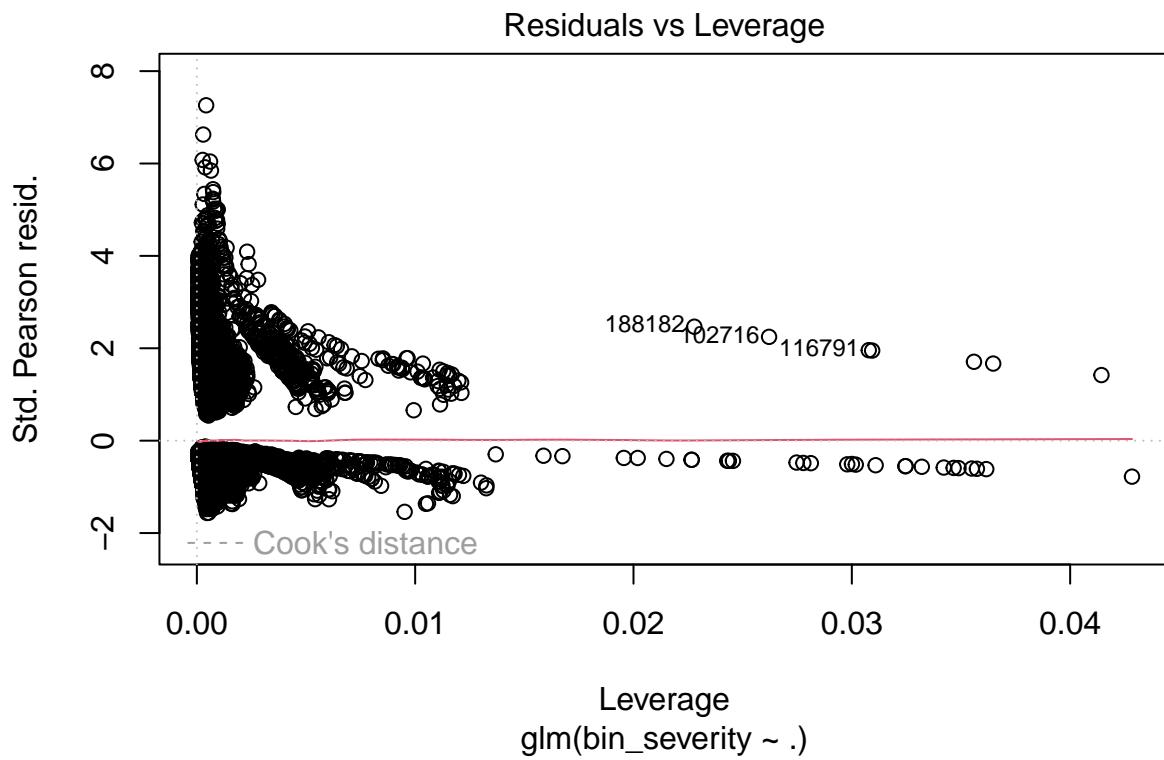
```

## age_band_of_casualty11      0.469714   0.057027   8.237 < 2e-16 ***
## time_group2                 -0.328354   0.039882  -8.233 < 2e-16 ***
## time_group3                 -0.544970   0.038565 -14.131 < 2e-16 ***
## time_group4                 -0.491883   0.037625 -13.073 < 2e-16 ***
## time_group5                 -0.450272   0.034891 -12.905 < 2e-16 ***
## time_group6                 -0.271676   0.035784  -7.592 3.15e-14 ***
## region2                     -0.036145   0.027132  -1.332 0.182797
## region3                     -0.291325   0.027280 -10.679 < 2e-16 ***
## region4                     -0.039928   0.037877  -1.054 0.291810
## region5                     0.003611    0.027774   0.130 0.896552
## region7                     0.615285    0.031638  19.448 < 2e-16 ***
## region8                     0.044661    0.024867   1.796 0.072501 .
## region9                     -0.267143   0.028760  -9.289 < 2e-16 ***
## region10                    -0.066668   0.037540  -1.776 0.075746 .
## region11                    -0.160294   0.029819  -5.376 7.63e-08 ***
## region12                    0.053446    0.028424   1.880 0.060061 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 187526  on 173397  degrees of freedom
## Residual deviance: 178926  on 173330  degrees of freedom
## AIC: 179062
##
## Number of Fisher Scoring iterations: 4
plot(fit)

```

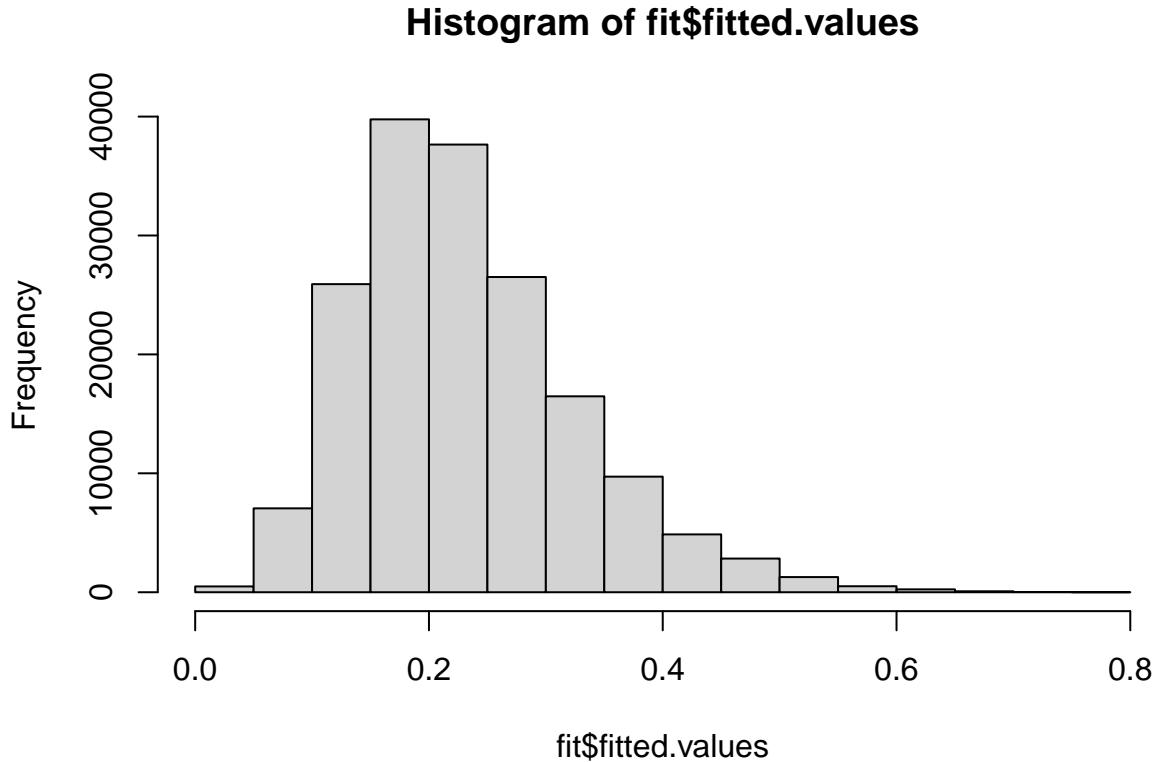






This histogram shows the distribution of fitted values from the training data.

```
hist(fit$fitted.values)
```

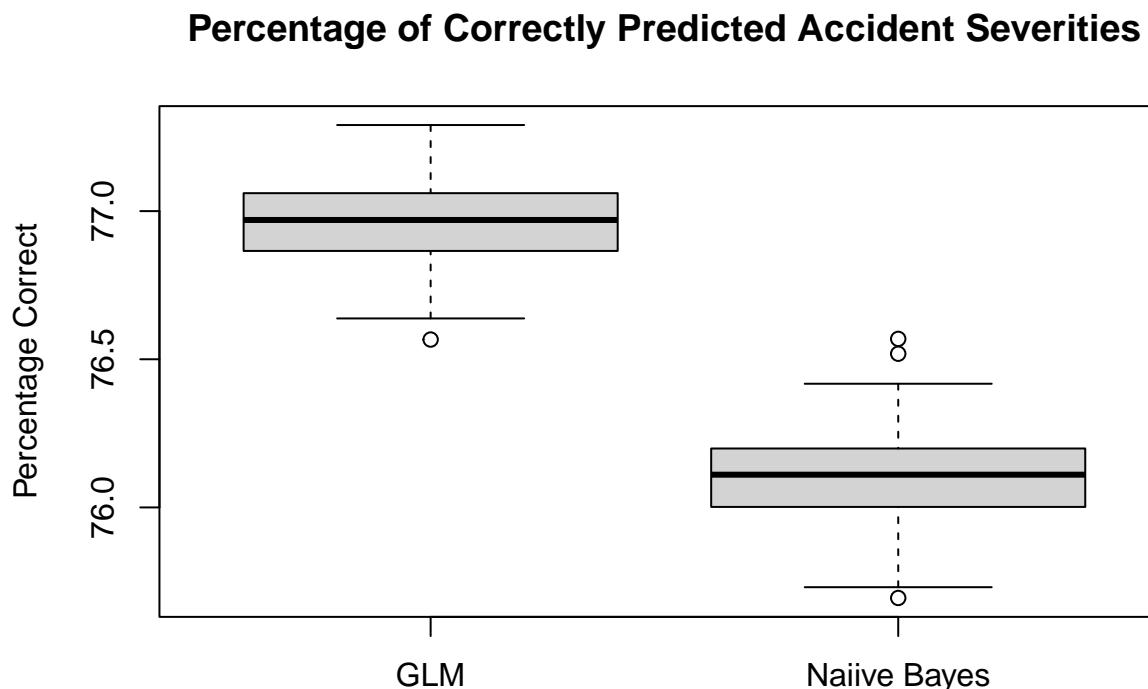


## Cross Validation & Model Refining

We can perform some bootstrapping on the model. Note that since this process can take a long time to run, it is wrapped in an if statement which checks whether or not the RDS file containing the accuracy dataframe exists first before running the bootstrapping process. If you want to rerun the bootstrapping, delete the accuracy.RDS file in the Data directory.

We can look at several useful plots to see how our model compares against a Naive-Bayes classifier.

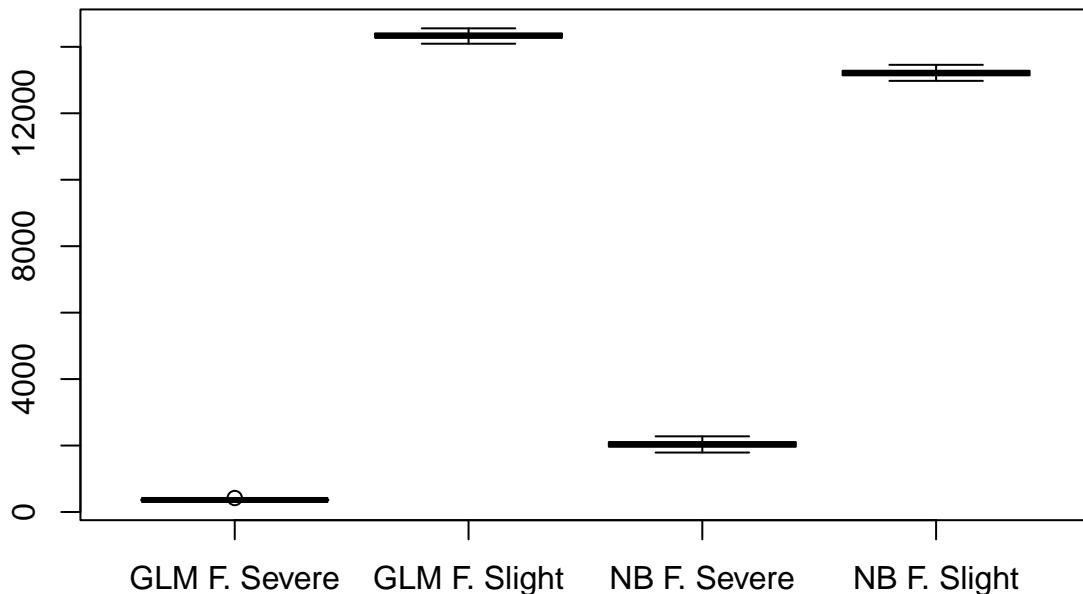
```
boxplot(accuracy$GLMPercCorrect, accuracy$NBPercCorrect,
        names = c("GLM", "Naïve Bayes"),
        ylab = "Percentage Correct",
        main = "Percentage of Correctly Predicted Accident Severities")
```



The plot below presents the proportion false predictions in each category. So a case when the model predicted severe but the accident was slight or visa-versa.

```
boxplot(accuracy$GLM.False.Severe, accuracy$GLM.False.Slight,
        accuracy$NB.False.Severe, accuracy$NB.False.Slight,
        names = c("GLM F. Severe", "GLM F. Slight",
                 "NB F. Severe", "NB F. Slight"),
        main = "Proportion of misclassifications"
)
```

## Proportion of misclassifications



Interestingly, the GLM model had a lower proportion of false severes than Naive-Bayes, but a higher proportion of false slight accidents than Naive-Bayes.

## Speed Limit and Road Type analysis

We now present a slightly more specific model than the one we have just seen. We include an interaction term between speed limit and road type, allowing us to assess whether or not speed limits on a given road type are appropriate based on the contribution they make towards accident severity.

```
fit_rt_sl <- glm(  
  data = df %>% select(-region),  
  formula = bin_severity ~ . + speed_limit*road_type,  
  family = binomial(link = "logit")  
)
```

We can view the summary of this model as before.

```
summary(fit_rt_sl)
```

```
##  
## Call:  
## glm(formula = bin_severity ~ . + speed_limit * road_type, family = binomial(link = "logit"),  
##       data = df %>% select(-region))  
##  
## Deviance Residuals:  
##      Min        1Q    Median        3Q       Max  
## -1.5573  -0.7434  -0.6308  -0.4466   2.8531  
##  
## Coefficients:  
##                               Estimate Std. Error z value Pr(>|z|)  
## (Intercept)                 -1.3795346  0.4354069 -3.168 0.001533 **  
## day_of_week2                -0.1645752  0.0230469 -7.141 9.27e-13 ***  
## day_of_week3                -0.1790344  0.0229455 -7.803 6.07e-15 ***
```

## day_of_week4	-0.1830192	0.0228015	-8.027	1.00e-15	***
## day_of_week5	-0.1128444	0.0224583	-5.025	5.04e-07	***
## day_of_week6	-0.1275460	0.0221069	-5.770	7.95e-09	***
## day_of_week7	-0.0298816	0.0226302	-1.320	0.186691	
## road_type2	-0.3066421	0.2389231	-1.283	0.199340	.
## road_type3	0.1710625	0.0960924	1.780	0.075045	.
## road_type6	-0.0036157	0.0876975	-0.041	0.967113	
## road_type7	0.0864231	0.1597606	0.541	0.588539	
## road_type9	-0.7932868	0.2253116	-3.521	0.000430	***
## speed_limit	0.1227718	0.0214133	5.733	9.84e-09	***
## light_conditions4	0.0548073	0.0199541	2.747	0.006020	**
## light_conditions5	0.0767796	0.0687500	1.117	0.264082	
## light_conditions6	0.2254801	0.0268825	8.388	< 2e-16	***
## light_conditions7	-0.4899482	0.0560870	-8.736	< 2e-16	***
## weather_conditions2	-0.1322569	0.0221871	-5.961	2.51e-09	***
## weather_conditions3	0.2334880	0.1440731	1.621	0.105099	
## weather_conditions4	0.1839108	0.0439573	4.184	2.87e-05	***
## weather_conditions5	-0.0045052	0.0440909	-0.102	0.918614	
## weather_conditions6	0.6206597	0.2087798	2.973	0.002951	**
## weather_conditions7	-0.0455868	0.0731385	-0.623	0.533091	
## weather_conditions8	-0.1134225	0.0397608	-2.853	0.004336	**
## weather_conditions9	-0.3455529	0.0548154	-6.304	2.90e-10	***
## road_surface_conditions2	-0.0087437	0.0166089	-0.526	0.598577	
## road_surface_conditions3	-1.0026694	0.1950145	-5.142	2.73e-07	***
## road_surface_conditions4	-0.2595558	0.0668642	-3.882	0.000104	***
## road_surface_conditions5	-0.1507025	0.1324134	-1.138	0.255070	
## road_surface_conditions9	-0.2546095	0.1367278	-1.862	0.062580	.
## sex_of_driver2	-0.1600455	0.0143870	-11.124	< 2e-16	***
## sex_of_driver3	-0.5051989	0.0598783	-8.437	< 2e-16	***
## age_band_of_driver2	-0.1828661	0.4571329	-0.400	0.689136	
## age_band_of_driver3	0.0645980	0.4331327	0.149	0.881442	
## age_band_of_driver4	0.0807948	0.4284558	0.189	0.850428	
## age_band_of_driver5	0.0897807	0.4281940	0.210	0.833923	
## age_band_of_driver6	0.0732885	0.4279359	0.171	0.864019	
## age_band_of_driver7	0.0702542	0.4279984	0.164	0.869616	
## age_band_of_driver8	0.1318966	0.4280616	0.308	0.757987	
## age_band_of_driver9	0.2163007	0.4281824	0.505	0.613446	
## age_band_of_driver10	0.2058689	0.4286267	0.480	0.631015	
## age_band_of_driver11	0.1048349	0.4292019	0.244	0.807033	
## sex_of_casualty2	-0.3434573	0.0134682	-25.501	< 2e-16	***
## age_band_of_casualty2	-0.0982561	0.0639893	-1.536	0.124659	
## age_band_of_casualty3	0.0795458	0.0586798	1.356	0.175229	
## age_band_of_casualty4	-0.0811427	0.0524944	-1.546	0.122167	
## age_band_of_casualty5	-0.2230167	0.0515831	-4.323	1.54e-05	***
## age_band_of_casualty6	-0.2560578	0.0497382	-5.148	2.63e-07	***
## age_band_of_casualty7	-0.2035090	0.0504294	-4.036	5.45e-05	***
## age_band_of_casualty8	-0.1152375	0.0506194	-2.277	0.022813	*
## age_band_of_casualty9	0.0626471	0.0515696	1.215	0.224440	
## age_band_of_casualty10	0.2868436	0.0546039	5.253	1.50e-07	***
## age_band_of_casualty11	0.4433291	0.0568890	7.793	6.55e-15	***
## time_group2	-0.3218892	0.0397006	-8.108	5.15e-16	***
## time_group3	-0.5323830	0.0383671	-13.876	< 2e-16	***
## time_group4	-0.4728361	0.0374291	-12.633	< 2e-16	***
## time_group5	-0.4354031	0.0346962	-12.549	< 2e-16	***

```

## time_group6      -0.2710653  0.0355991  -7.614 2.65e-14 ***
## road_type2:speed_limit   0.1613296  0.0835182   1.932 0.053400 .
## road_type3:speed_limit  -0.0006004  0.0229148  -0.026 0.979096
## road_type6:speed_limit   0.1339845  0.0220349   6.081 1.20e-09 ***
## road_type7:speed_limit   0.0065968  0.0329420   0.200 0.841280
## road_type9:speed_limit   0.1006326  0.0677592   1.485 0.137504
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 187526  on 173397  degrees of freedom
## Residual deviance: 179855  on 173335  degrees of freedom
## AIC: 179981
##
## Number of Fisher Scoring iterations: 4

```

Notice the coefficients for road type 2 and 6 (one way street, single carriageway respectively) are on the same order as speed limit overall. This would suggest that perhaps a lower speed limit would be appropriate on these particular road types.