INFO Challenge

```
drivers<-read.csv("died_survival.csv")
head(drivers)
```

	year	case	par	repjur	crash_dt	crash_tm	accday	accmon	holiday ▸
	<int>	<int>	<chr>	<int>	<chr>	<chr>	<int>	<int>	<int>
1	2017	1	E628946	2	1/1/2017	2:12	1	1	1
2	2017	2	E627989	26	1/2/2017	17:14	2	1	1
3	2017	4	3747633	263	1/1/2017	18:47	1	1	1
4	2017	5	E628691	4	1/1/2017	3:50	1	1	1
5	2017	6	3746306	263	1/5/2017	9:53	5	1	0
6	2017	6	3746306	263	1/5/2017	9:53	5	1	0

6 rows | 1-10 of 301 columns

```
df<-data.frame(drivers)
head(df)
```

	year	case	par	repjur	crash_dt	crash_tm	accday	accmon	holiday ▸
	<int>	<int>	<chr>	<int>	<chr>	<chr>	<int>	<int>	<int>
1	2017	1	E628946	2	1/1/2017	2:12	1	1	1
2	2017	2	E627989	26	1/2/2017	17:14	2	1	1
3	2017	4	3747633	263	1/1/2017	18:47	1	1	1
4	2017	5	E628691	4	1/1/2017	3:50	1	1	1
5	2017	6	3746306	263	1/5/2017	9:53	5	1	0
6	2017	6	3746306	263	1/5/2017	9:53	5	1	0

6 rows | 1-10 of 301 columns

```
#FOR LOGISTIC REGRESSION
df$DEATH<-factor(df$DEATH, levels=c("Survived", "Died"))
df$age <- as.numeric(df$age)
df$sex <- factor(df$sex)#, exclude=c("8", "9"))
df$dr_drug<- factor(df$dr_drug, levels=c(0, 1))
df$dr_drink <- factor(df$dr_drink, levels=c("0", "1"))
df$dr_imp <- factor(df$dr_imp, levels=c("0", "1"))
df$dr_spd <- factor(df$dr_spd, levels=c("0", "1"))
df$dr_unlic <- factor(df$dr_unlic, levels=c("0", "1"))
df$is_resident <- factor(df$is_resident, levels=c(TRUE, FALSE))
df$weather <- factor(df$weather)#, exclude = c("8", "98", "99"))
df$surfcond <- factor(df$surfcond)#, exclude = c(0, 8, 9))
df$seatbelt <- factor(df$seatbelt, levels=c("Yes", "No"))
df$lightcond <- factor(df$lightcond)#, exclude = c("7", "8", "9"))
df$criticaleventcat <- factor(df$criticaleventcat)#, exclude = c("9"))
```

```
#FOR LINEAR REGRESSION
#df$DEATH<-as.numeric(df$DEATH)
#df$dr_drug<- as.numeric(df$dr_drug)
#df$dr_drink <- as.numeric(df$dr_drink)
```

```
fit1 <- glm(DEATH ~ sex + age + dr_drug + dr_drink + dr_imp + dr_spd + dr_unlic + is_resident +
weather + surfcond + seatbelt + lightcond + criticaleventcat, data=df, family="binomial")
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(fit1)
```

```
##
## Call:
## glm(formula = DEATH ~ sex + age + dr_drug + dr_drink + dr_imp +
##     dr_spd + dr_unlic + is_resident + weather + surfcond + seatbelt +
##     lightcond + criticaleventcat, family = "binomial", data = df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3174  -0.6972  -0.4494   0.7795   2.4615
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)       -8.297e-01  4.701e-01  -1.765 0.077576 .
## sex2               5.900e-02  9.765e-02   0.604 0.545738
## sex8              -8.454e-01  3.832e-01  -2.206 0.027384 *
## sex9              -1.590e+01  7.497e+02  -0.021 0.983084
## age                1.201e-04  7.532e-04   0.159 0.873309
## dr_drug1          -1.404e-01  1.660e-01  -0.845 0.397880
## dr_drink1          5.895e-02  1.220e-01   0.483 0.628970
## dr_imp1            1.619e+00  1.831e-01   8.844  < 2e-16 ***
## dr_spd1            3.832e-01  1.037e-01   3.695 0.000220 ***
## dr_unlic1         -3.878e-01  1.096e-01  -3.539 0.000401 ***
## is_residentFALSE   5.181e-02  9.611e-02   0.539 0.589851
## weather2           2.489e-02  2.017e-01   0.123 0.901780
## weather3          -3.992e-01  7.898e-01  -0.505 0.613229
## weather4          -4.545e-01  4.976e-01  -0.913 0.361016
## weather5           3.241e-01  2.609e-01   1.242 0.214115
## weather6           3.583e+00  1.189e+00   3.015 0.002572 **
## weather7           1.566e+00  1.433e+00   1.093 0.274417
## weather8          -1.536e+00  9.375e-01  -1.639 0.101292
## weather10         -1.643e-02  1.260e-01  -0.130 0.896208
## weather98         -9.078e-01  1.290e+00  -0.704 0.481539
## weather99          1.379e+00  1.203e+00   1.146 0.251731
## surfcond1         -2.042e-01  4.483e-01  -0.456 0.648712
## surfcond2         -2.046e-01  4.689e-01  -0.436 0.662632
## surfcond3         -7.811e-02  7.813e-01  -0.100 0.920368
## surfcond4         -1.135e-01  5.219e-01  -0.217 0.827906
## surfcond6          1.229e+00  8.342e-01   1.473 0.140776
## surfcond8          2.883e-01  7.735e-01   0.373 0.709397
## surfcond10        -1.433e-01  7.021e-01  -0.204 0.838323
## surfcond11        -6.987e-01  8.446e-01  -0.827 0.408075
## surfcond98         1.594e+00  1.355e+00   1.177 0.239265
## surfcond99         8.236e-01  8.996e-01   0.916 0.359913
## seatbeltNo         6.010e-01  8.451e-02   7.112 1.14e-12 ***
## lightcond2         1.044e-01  1.077e-01   0.969 0.332617
## lightcond3        -2.092e-01  1.168e-01  -1.791 0.073248 .
## lightcond4         2.050e-01  2.587e-01   0.793 0.428042
## lightcond5         7.060e-02  2.211e-01   0.319 0.749458
## lightcond6        -2.736e-01  7.866e-01  -0.348 0.727947
## lightcond7        -1.949e+01  6.523e+03  -0.003 0.997616
## lightcond8        -8.461e-01  1.302e+00  -0.650 0.515925
## lightcond9         1.239e+00  6.437e-01   1.925 0.054218 .
## criticaleventcat2 -2.672e-01  1.332e-01  -2.006 0.044875 *
## criticaleventcat3 -9.196e-01  1.636e-01  -5.620 1.91e-08 ***
## criticaleventcat4 -1.040e+00  1.561e-01  -6.660 2.75e-11 ***
## criticaleventcat5 -1.798e+01  2.524e+02  -0.071 0.943233
## criticaleventcat6 -6.048e-01  3.934e-01  -1.538 0.124151
## criticaleventcat7 -1.854e+00  2.899e-01  -6.395 1.60e-10 ***
## criticaleventcat9 -8.239e-01  1.045e+00  -0.788 0.430672
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5106.7  on 4131  degrees of freedom
## Residual deviance: 3704.7  on 4085  degrees of freedom
##   (5 observations deleted due to missingness)
## AIC: 3798.7
##
## Number of Fisher Scoring iterations: 17
```

```
set.seed(23457)
#Use 70% of dataset as training set and remaining 30% as testing set
sample <- sample(nrow(df), 0.7*nrow(df))
#sample <- sample(c(TRUE, FALSE), nrow(df), replace=TRUE, prob=c(0.6,0.4))
train  <- df[sample, ]
test   <- df[-sample, ]

#view dimensions of training set
dim(train)
```

```
## [1] 2895  300
```

```
#view dimensions of test set
dim(test)
```

```
## [1] 1242  300
```

Using a cutoff of 0.5 and computing the confusion matrix for IN-SAMPLE PREDICTIONS

```
cutoff <- 0.5
ActualTrain <- train$DEATH
prediction.train <- predict(fit1,newdata = train, type="response")
PredictedTrain <- ifelse(prediction.train>cutoff,"Died","Survived")
PredictedTrain <- factor(PredictedTrain,levels=c("Survived","Died"))
confusionTrain<-table(ActualTrain, PredictedTrain)  #CONFUSION MATRIX FOR IN-SAMPLE PREDICTIONS
confusionTrain
```

```
##            PredictedTrain
## ActualTrain Survived Died
##    Survived     1710  285
##    Died          398  499
```

Using a cutoff of 0.5 and computing the confusion matrix for OUT OF SAMPLE PREDICTIONS

```
cutoff <- 0.5
ActualTest <- test$DEATH
prediction.test <- predict(fit1,newdata = test, type="response")
PredictedTest <- ifelse(prediction.test>cutoff,"Died","Survived")
PredictedTest <- factor(PredictedTest,levels=c("Survived","Died"))
confusionTest<-table(ActualTest, PredictedTest)  #CONFUSION MATRIX FOR OUT-OF-SAMPLE PREDICTIONS
confusionTest
```

```
##           PredictedTest
## ActualTest Survived Died
##   Survived      752  110
##   Died          168  210
```

```
#Training sensitivity
(SensitivityTrain <- confusionTrain[2,2]/sum(confusionTrain[2,]))
```

```
## [1] 0.5562988
```

```
#Training specificity
(SpecificityTrain <- confusionTrain[1,1]/sum(confusionTrain[1,]))
```

```
## [1] 0.8571429
```

```
#Training PPV
(PPVTrain <- confusionTrain[2,2]/sum(confusionTrain[,2]))
```

```
## [1] 0.6364796
```

```
#Training NPV
(NPVTrain <- confusionTrain[1,1]/sum(confusionTrain[,1]))
```

```
## [1] 0.8111954
```

```
#Test sensitivity
(SensitivityTest <- confusionTest[2,2]/sum(confusionTest[2,]))
```

```
## [1] 0.5555556
```

```r
#Test specificity
(SpecificityTest <- confusionTest[1,1]/sum(confusionTest[1,]))
```

```
## [1] 0.8723898
```

```r
#Test PPV
(PPVTest <- confusionTest[2,2]/sum(confusionTest[,2]))
```

```
## [1] 0.65625
```

```r
#Test NPV
(NPVTest <- confusionTest[1,1]/sum(confusionTest[,1]))
```

```
## [1] 0.8173913
```