Univerzitet u Sarajevu

Elektrotehnički fakultet Sarajevo

Računarstvo i informatika

ZADAĆA 2: Izvještaj

iz predmeta Mašinsko učenje

Sara Ramadanović

Sarajevo, 2022. godina

Zadatak 1

U sklopu zadaće 2 iz predmeta Mašinsko učenje koristit će se set podataka iz zadaće 1 u kojoj je izvršeno predprocesiranje podataka i izvršene su radnje potrebne za daljnje korištenje podataka u svrhu treniranja modela i evaluacije. Dataset se vodi pod nazivom *dataset.csv*.

U zadatku 1 bilo je potrebno kreirati modele, te kreirane modele provjeriti i istrenirati. Da bismo postigli željene rezultate, potrebno je prvo ulazni set podataka podijeliti na trening i testni skup podataka i to je potrebno uraditi prije svakog kreiranog modela. Trening skup podataka se vodi pod nazivom *podaci_train* i čini ga 80% podataka, a testni skup podataka se vodi pod nazivom *podaci_test* i čini ga ostatak od 20% podataka iz izvornog seta podataka.

Logistička regresia

Prvi model koji je kreiran jeste model logističke regresije i model je kreiran pomoću funkcije glm(). Međutim radi pripreme podataka za logističku regresiju, prije svega bilo je potrebno prebaciti vrijednosti iz numeričkih, kao i kategorisanje.

```
#Logisticka regresija
```{r}

logitMod <- glm(RainTomorrow~Location+MinTemp+MaxTemp+Rainfall+WindGustDir+WindGustSpeed+
                  RainToday+Humidity9am+Cloud9am+Pressure9am+WindDir9am+WindDir3pm,family=binomial(link='logit'),
data=podaci_train, na.action="na.omit")

predictedLog <- predict(logitMod, podaci_test, type="response")

logitMod

```
```

Pozivanjem logitMod dobijemo prikaz modela sa sljedecim informacijama:

koeficijanti, standardne greške, z-statistika kao i stupnjevi slobode, odstupanje od nule, rezidualna devijacija i AIC.

```
> logitMod

Call:  glm(formula = RainTomorrow ~ Location + MinTemp + MaxTemp + Rainfall +
    WindGustDir + WindGustSpeed + RainToday + Humidity9am + Cloud9am +
    Pressure9am + WindDir9am + WindDir3pm, family = binomial(link = "logit"),
    data = podaci_train, na.action = "na.omit")

Coefficients:
             (Intercept)      LocationAliceSprings          LocationBallarat           LocationBendigo
                74.94092                  -0.49556                   0.04838                  -0.17150
        LocationBrisbane           LocationCairns         LocationCanberra             LocationCobar
                 0.56888                   0.26345                   0.26872                  -0.10325
    LocationCoffsHarbour           LocationDarwin           LocationHobart        LocationKatherine
                 0.61430                   0.32595                   0.07772                   0.19355
       LocationMelbourne  LocationMelbourneAirport          LocationMildura            LocationMoree
                -0.14738                  -0.83337                  -0.15539                  -0.39152
    LocationMountGambier     LocationNorfolkIsland        LocationNuriootpa       LocationPearceRAAF
                -0.54029                   0.02517                  -0.15672                   0.39359
          LocationPerth      LocationPerthAirport          LocationPortland         LocationRichmond
                 0.01470                  -0.28861                  -0.02263                  -0.72472
           LocationSale           LocationSydney     LocationSydneyAirport       LocationTownsville
                -0.50229                  -0.52955                  -0.03473                  -0.63504
          LocationUluru        LocationWaggaWagga         LocationWatsonia      LocationWilliamtown
                 0.05784                  -0.60666                  -0.63540                  -1.18913
      LocationWollongong          LocationWoomera                   MinTemp                   MaxTemp
                 0.24526                  -1.34897                   0.06958                  -0.09624
              Rainfall               WindGustDir           WindGustSpeed                 RainToday
                -0.01275                   0.01228                   0.02644                   0.33495
            Humidity9am                  Cloud9am              Pressure9am                WindDir9am
                 0.01622                   0.21273                  -0.07660                  -0.06955
             WindDir3pm
                 0.03197

Degrees of Freedom: 1599 Total (i.e. Null);  1555 Residual
Null Deviance:      1726
Residual Deviance: 1374         AIC: 1464
>
```

Međutim radi pripreme podataka za logističku regresiju, prije svega bilo je potrebno prebaciti vrijednosti iz numeričkih, kao i kategorisanje.

```
#transformacija
```{r}

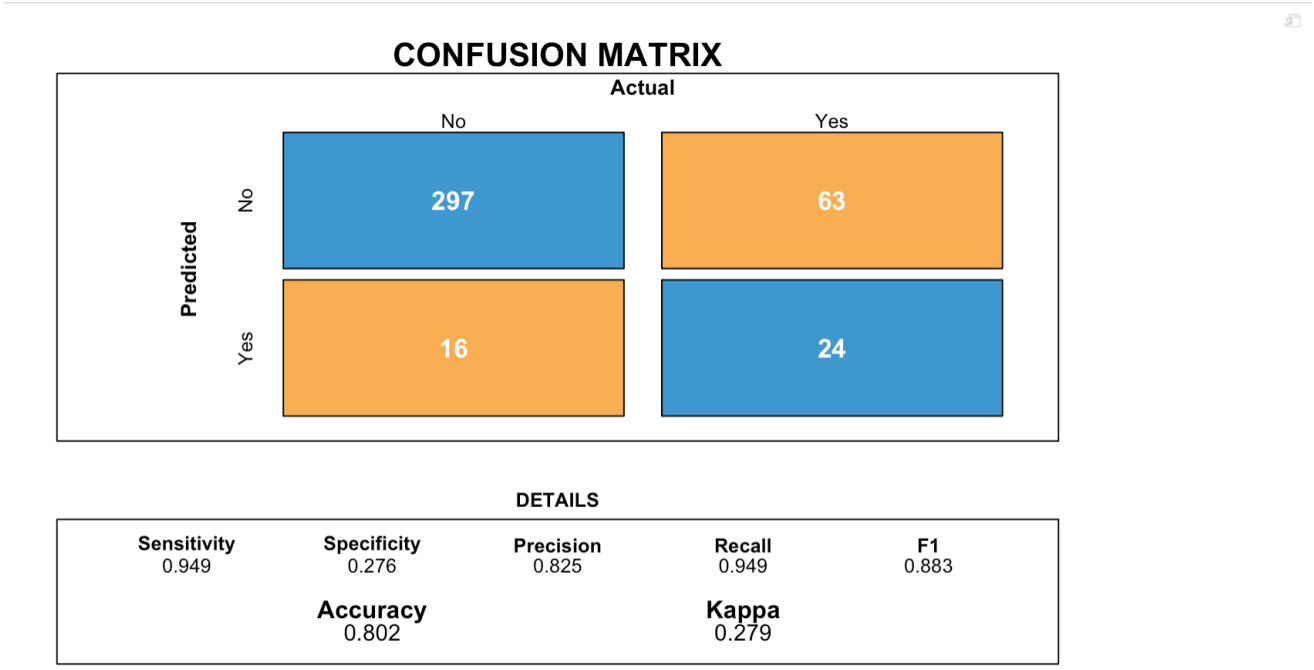
pred_logit2 <- predict(logitMod, newdata = podaci_test)
pred_logit <- ifelse(pred_logit2 > 0, 1, 0)
pred_logit_cat <- unname(pred_logit)
pred_logit_cat<-ifelse(pred_logit_cat==1,"Yes","No")
pred_logit_cat <- as.factor(pred_logit_noname)

pred_logit_cat
```
```

Ovim pozivima, postavljeno je ukoliko je vrijednost veća od 0, oznaka je "Yes", a ukoliko je vrijednost manja od 0, oznaka će biti "No".

```
> pred_logit_cat
  [1] No  No  No  No  No  No  No  Yes No  No  No  No  No  No  No  No  No  No  No  No  Yes No  No  No  No  No  No  No
 [29] No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  Yes No  No  No  No  No  No
 [57] No  Yes No  No  No  No  No  No  No  Yes No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No
 [85] No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No
[113] No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  Yes No  No  No  No
[141] No  No  No  Yes Yes Yes No  Yes No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No
[169] No  No  No  No  No  No  No  No  No  No  Yes No  No  No  Yes No  No  No  No  No  No  Yes No  No  No  No  No  No
[197] No  No  No  No  Yes No  No  No  No  No  Yes No  No  No  Yes No  No  No  No  No  No  No  No  No  Yes No  No  No
[225] No  No  No  No  No  Yes No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  Yes No
[253] No  No  No  Yes No  No  No  No  No  No  No  No  No  No  No  No  No  Yes Yes Yes No  No  No  No  No  No  No  No
[281] No  No  No  No  No  Yes No  No  No  No  No  No  No  No  Yes No  No  No  No  No  Yes No  No  Yes No  No  No  No
[309] No  Yes No  No  No  Yes No  No  No  No  No  Yes No  No  No  Yes No  No  No  No  No  No  No  No  No  No  No  No
[337] No  No  No  No  No  No  No  No  No  No  Yes Yes No  No  Yes No  No  No  No  No  No  Yes No  No  Yes Yes No  No
[365] No  No  No  Yes No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  No  Yes No  No
[393] Yes No  No  No  No  No  No  No
Levels: No Yes
>
```

Nakon transformacije varijabli, potrebno je prikazati rezultate modela pomoću konfuzijske matrice.



Ako tumačimo rezultate modela po tačnošću, možemo reći da smo dobili poprilično zadovoljavajuću tačnost od 0.802, međutim ostali faktori nisu baš pokazali dobre rezultate s obzirom da je specifičnost niska i iznosi 0.276. Kappa je također prilično niska.

Sljedeći postupak na redu jeste prikaz ROC krive i to je urađeno na sljedeći način:

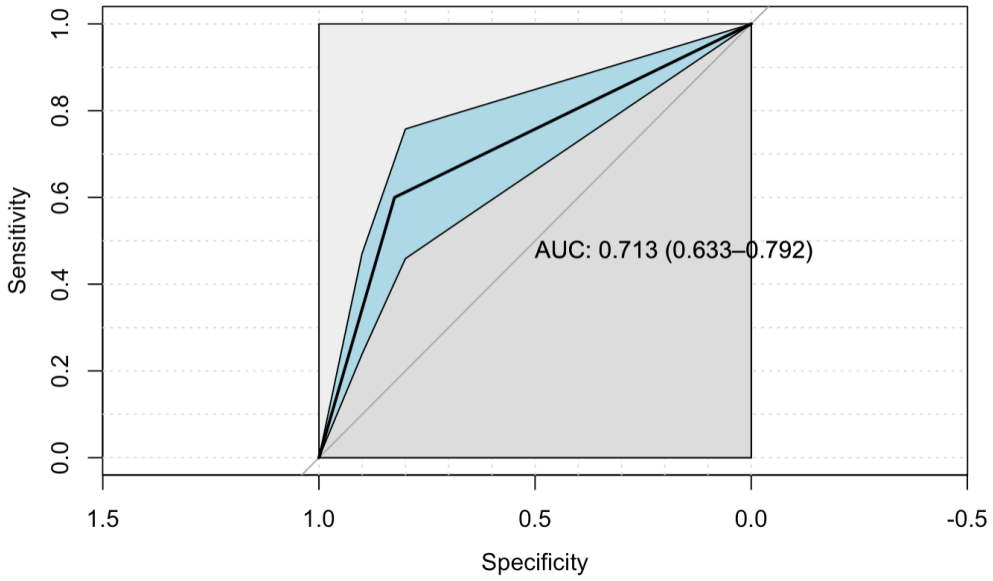
```r
#ROC
```{r}
logit_model_num <- as.numeric(pred_logit_cat)
podaci_test_num <- as.numeric(podaci_test$RainTomorrow)

lrROC <- roc(logit_model_num ~ podaci_test_num,smoothed = TRUE,
             # arguments for ci
             ci=TRUE, boot.n=100, ci.alpha=0.9, stratified=FALSE,
             # arguments for plot
             plot=TRUE, auc.polygon=TRUE, max.auc.polygon=TRUE, grid=TRUE,
             print.auc=TRUE, show.thres=TRUE)

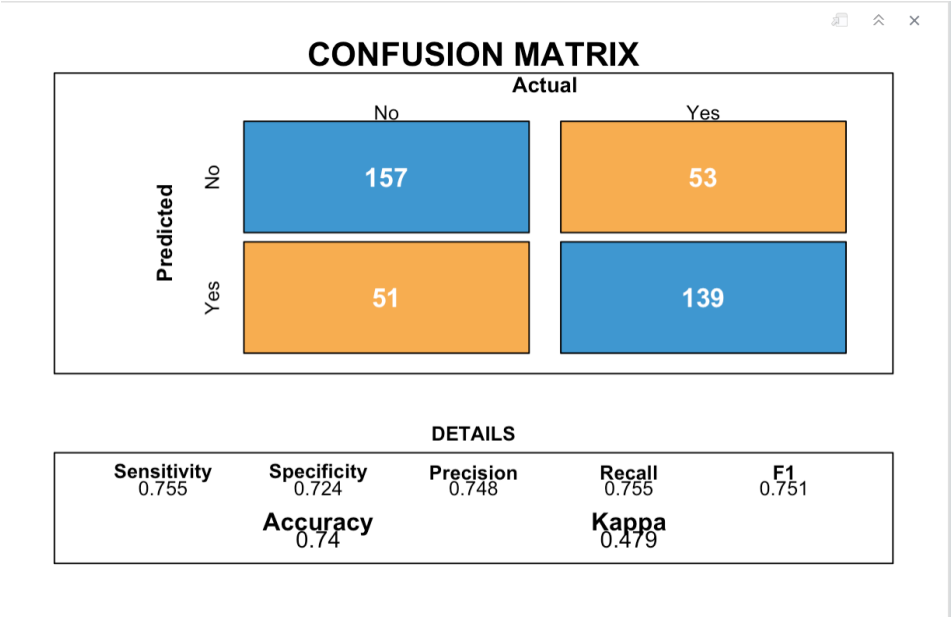
sens.ci <- ci.se(lrROC)
plot(sens.ci, type="shape", col="lightblue")

confusionMatrix(pred_logit_cat, podaci_test$RainTomorrow)

```
```



Nakon ovakvih rezultata odrađen je oversampling pozivanjem *ovun.sample()* metode korištenjem (method="both") i ta metoda je dala poprilično popravljene rezultate u odnosu na prethodnu konfuzijsku matricu. Prvo sam pokušala sa odvojenim oversamplingom i undersamplingom, međutim taj način nije dao pretjerano poboljšane rezultate, te sam se odlučila ipak za ovu metodu sa nasumičnim over i under samplingom.



Konfuzijska matrica nakon oversamplinga pokazuje dosta bolju osjetljivost i specifičnost, kao i kappa statistiku koja sada iznosi blizu 0.5.

Nakon oversamplinga korištena je k-fold validacija sa vrijednostima k=10 i k=5 i dobiveni su sljedeći rezultati:

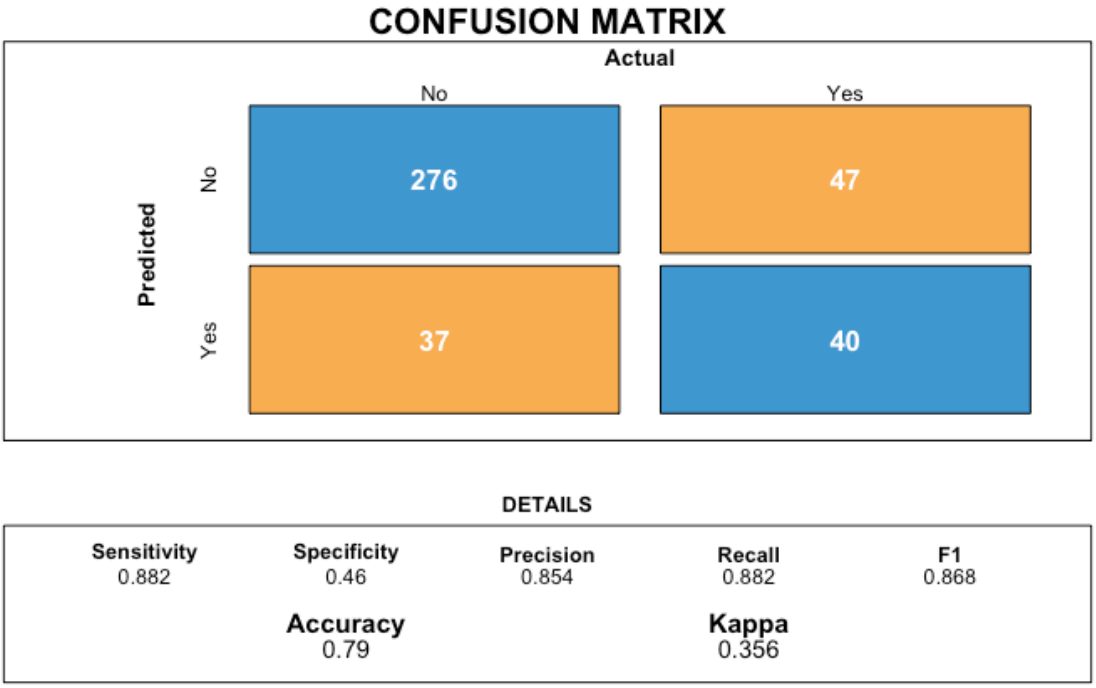
```
10-fold validacija
Najveća tačnost: 0.795 , fold: 7, najveća kappa: 0.4837572 , fold: 7
Najmanja tačnost: 0.725 , fold: 9, najmanja kappa: 0.3239013 , fold: 2
Srednja tačnost: 0.7515, srednja kappa: 0.3811166
```

```
5-fold validacija
Najveća tačnost: 0.7825 , fold: 2, najveća kappa: 0.4642197 , fold: 2
Najmanja tačnost: 0.71 , fold: 1, najmanja kappa: 0.321161 , fold: 1
Srednja tačnost: 0.7515, srednja kappa: 0.3820957
```

Naive Bayes

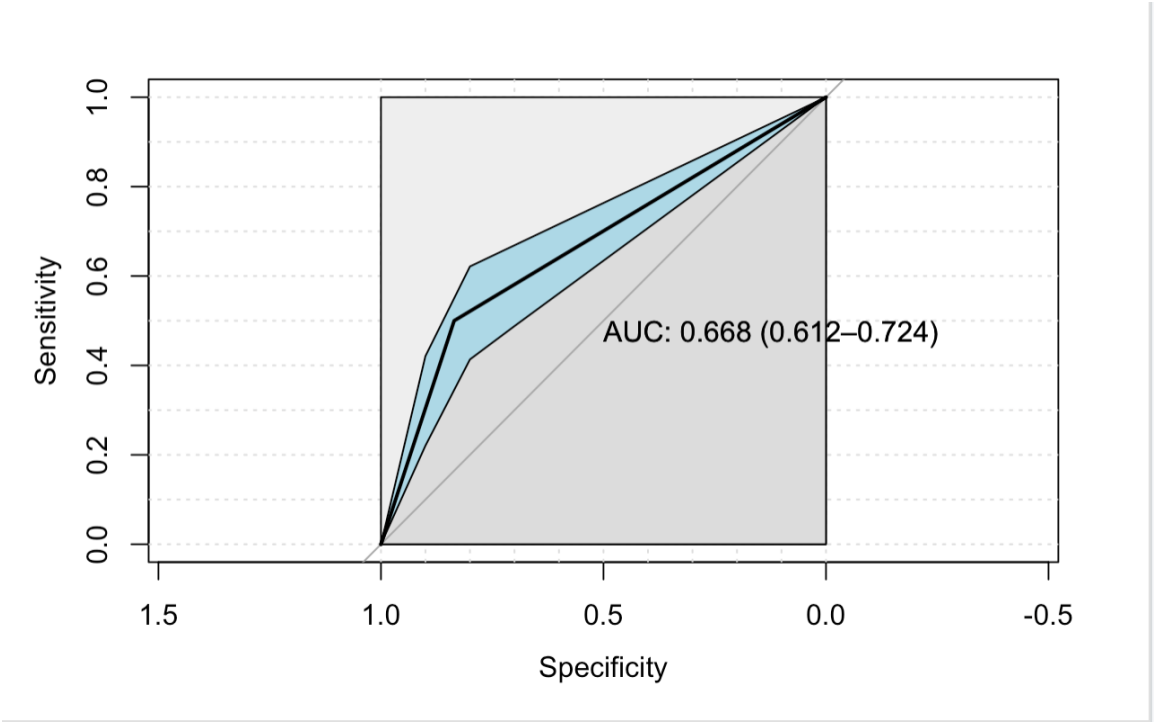
Sljedeći model koji je izgrađen u sklopu zadaće je Naivni Bayes koji koristi funkciju *naive_bayes* iz biblioteke *naivebayes.* Prvo je model izgrađen bez laplasijana, međutim nakon toga je dodan laplace=1 radi rješavanja problema nula vjerovatnoća (*zero probability*) što je generalno problem kod naivnog bayesa. Korištenjem laplasijana != 0, vjerovatnoća neće više biti 0.

Rezultat konfuzijske matrice neposredno nakon izgradnje modela je sljedeći:

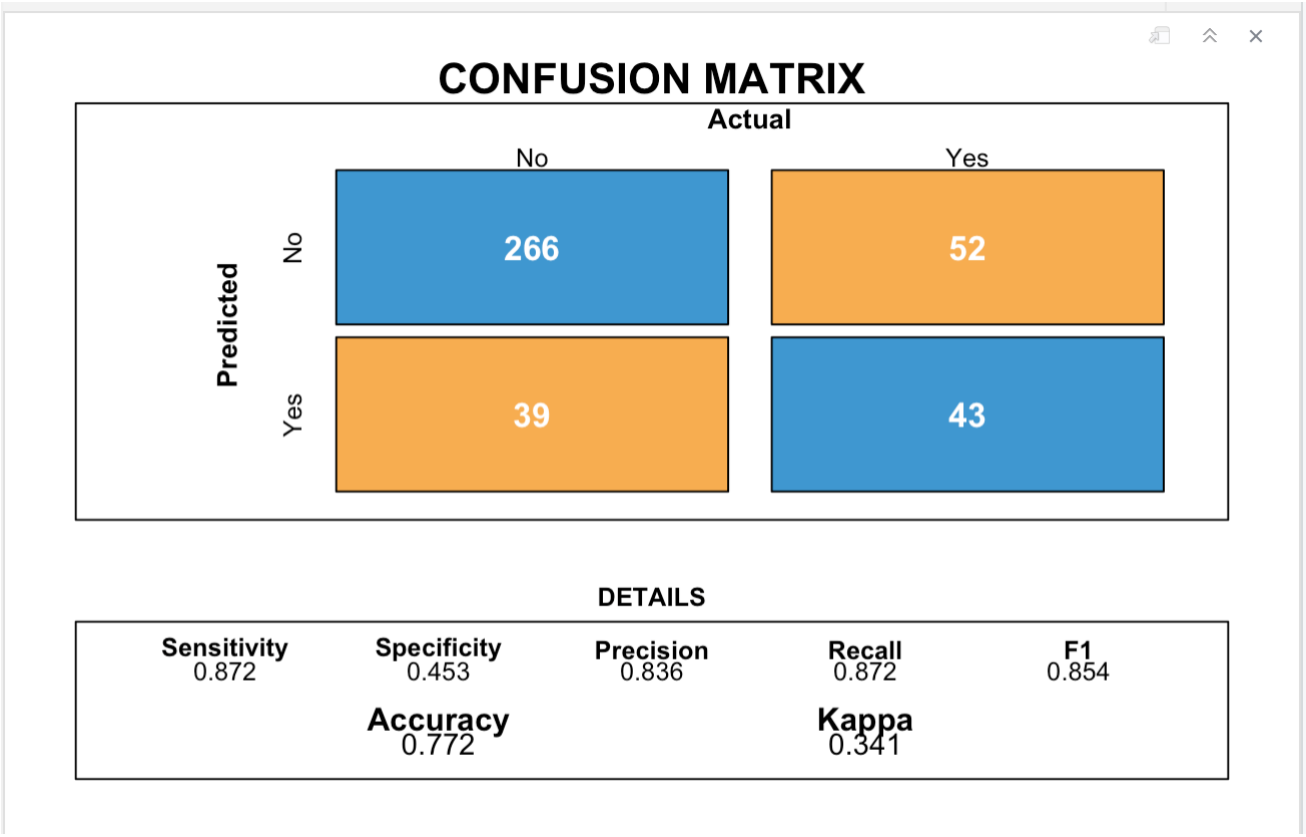


Dobili smo zadovoljavajuće rezultate gdje je tačnost nešto manja u odnosu na tačnost dobivenu odmah nakon izgradnje logističke regresije, međutim osjetljivost i specifičnost su nešto bolje u odnosu na logističku regresiju(prije popravljanja).

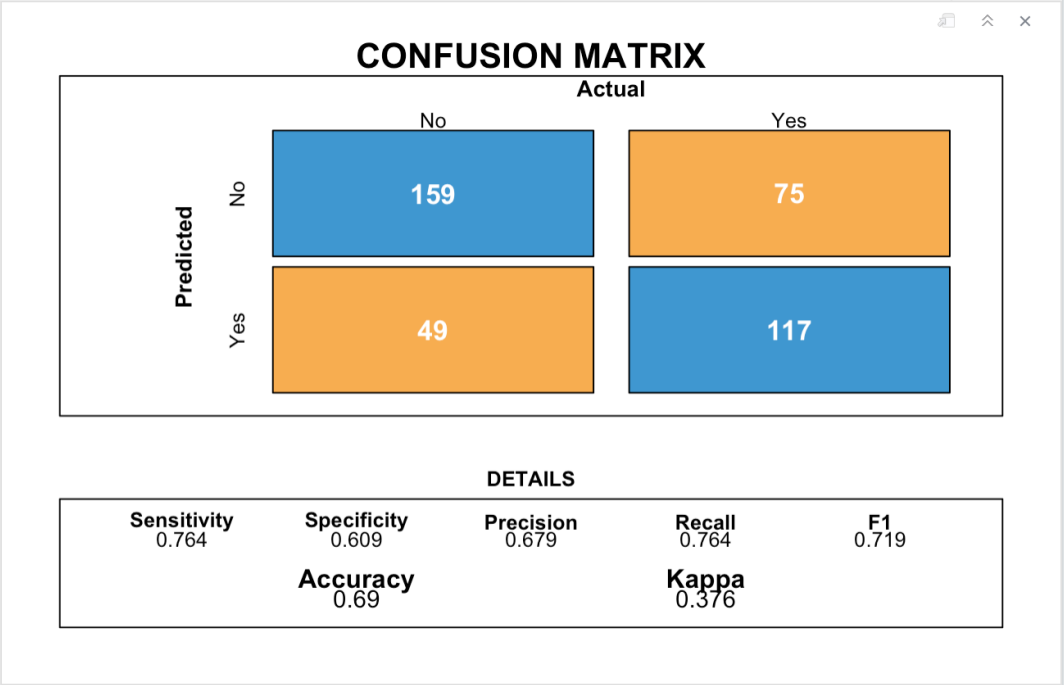
Slijedi nam prikaz ROC krive koju smo dobili istim postupkom kao i kod logističke regresije.



Korištenjem *ovun.sample()* sa *method="under"* dobijamo sljedeće rezultate. Tačnost je neznatno smanjena, a osjetljivost i specifičnost nisu pokazali neke pretjerane naznake poboljšanja.



Stoga, odlučila sam probati sa method="both" i tu već možemo vidjeti malo drugačije rezultate. Osjetljivost i specifičnost su sada u boljem omjeru, međutim, tačnost je ipak previše smanjena kao i preciznost.



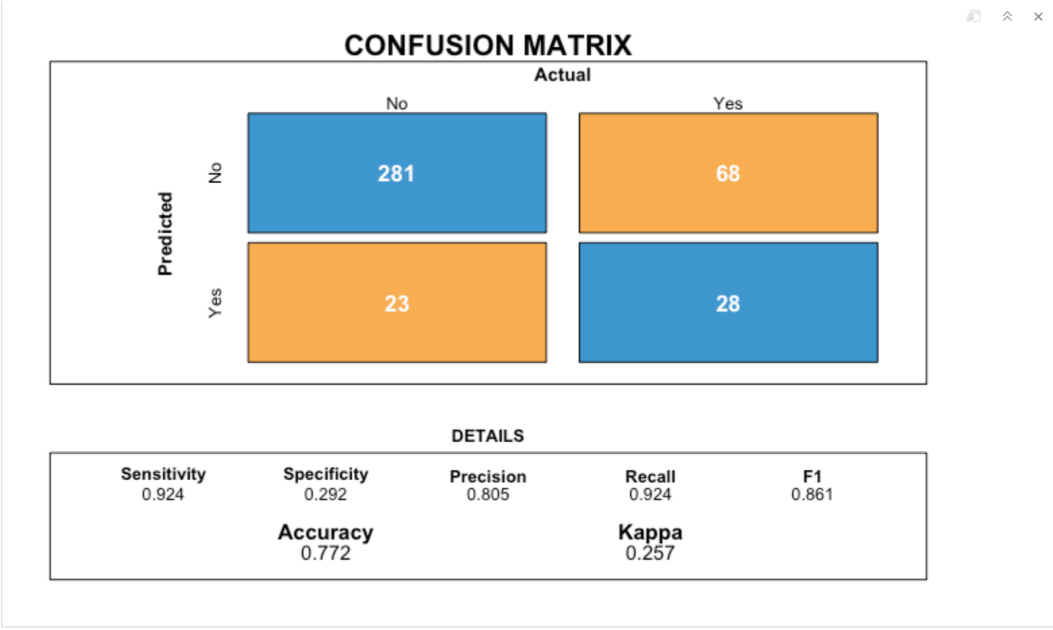
Kao i kod logističke regresije, i za bayesa je odrađena k-fold validacija.

```
10-fold validacija
Najveća tačnost: 0.8 , fold: 1, najveća kappa: 0.3972603 , fold: 7
Najmanja tačnost: 0.73 , fold: 2, najmanja kappa: 0.1882141 , fold: 10
Srednja tačnost: 0.7625, srednja kappa: 0.3063222
```

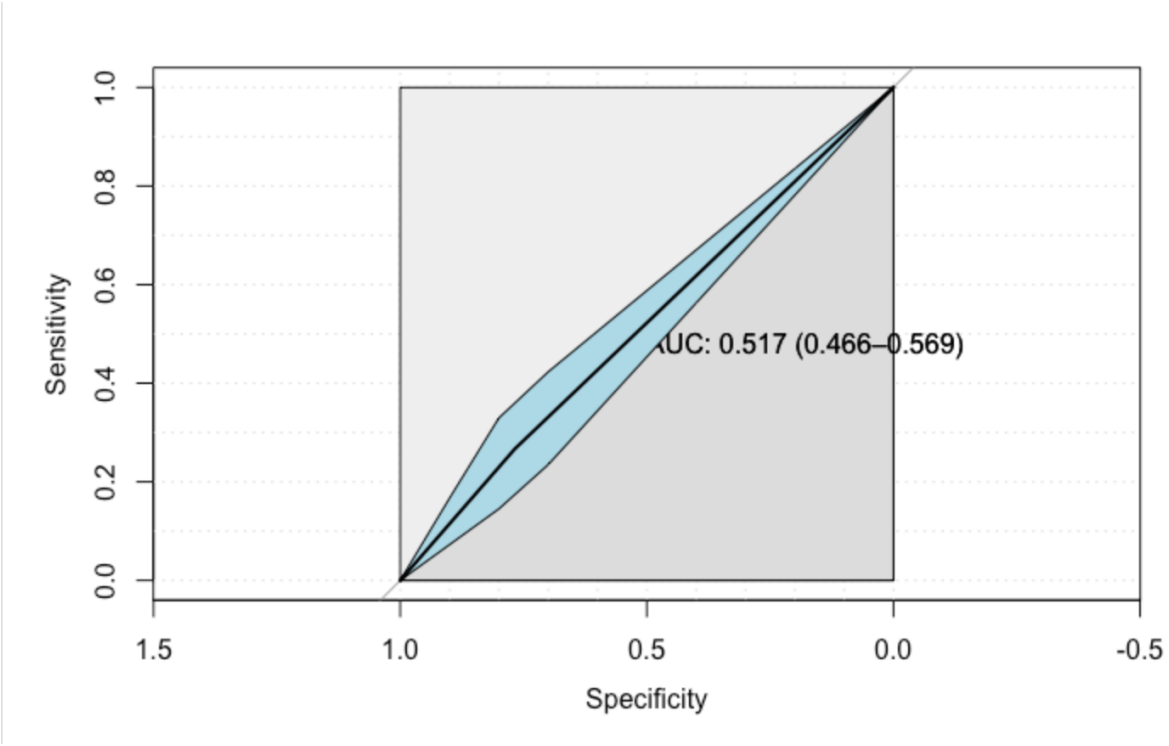
```
5-fold validacija
Najveća tačnost: 0.7675 , fold: 1, najveća kappa: 0.3468893 , fold: 4
Najmanja tačnost: 0.75 , fold: 5, najmanja kappa: 0.2570304 , fold: 5
Srednja tačnost: 0.7595, srednja kappa: 0.2950915
```

KNN

Za treniranje Knn modela prvo je bilo potrebno prebaciti sve kategoričke varijable u numeričke. Nakon pripreme varijabli, set podataka je podijeljen na trening i testni skup kao i u predhodna dva modela. Knn model kreiran je uz pomoć *knn3()* funkcije i dobiveni su sljedeći rezultati.

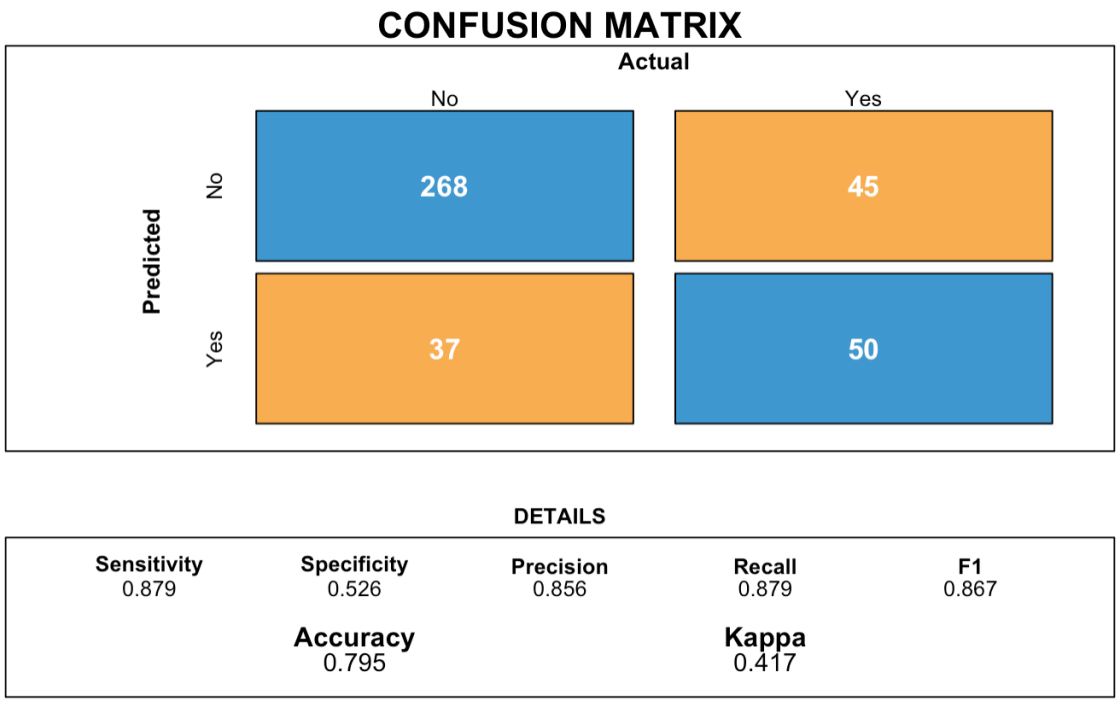


Pored konfuzijske matrice, ROC kriva za Knn izgleda ovako:



Na izvorno ostvarene rezultate, primjenjen je oversampling na više načina kako bi se ostvarili što bolji rezultati. Prvo sam pokušala sa metodom *ovun.sample(), method=”both”,* koja se nije loše pokazala, međutim htjela sam probati još poboljšati rezultate.

Najbolja verzija *ovun.sample()* metode bila je sa *method=”over”* i rezultati su sljedeći:



Popravljena je tačnost koja iznosi skoro 0.8, i razlika između osjetljivosti i specifičnosti je malo popravljena u odnosu na izvorni rezultat. Kappa je također poboljšana.

Nakon oversamplinga, slijedi validacija k-fold i rezultati za k=10 i k=5 su sljedeći:

```
10-fold validacija
Najveća tačnost: 0.8 , fold: 7, najveća kappa: 0.4342672 , fold: 5
Najmanja tačnost: 0.735 , fold: 4, najmanja kappa: 0.1643015 , fold: 4
Srednja tačnost: 0.7705, srednja kappa: 0.3035124

5-fold validacija
Najveća tačnost: 0.795 , fold: 1, najveća kappa: 0.3657807 , fold: 5
Najmanja tačnost: 0.74 , fold: 2, najmanja kappa: 0.2221391 , fold: 2
Srednja tačnost: 0.7705, srednja kappa: 0.3121294
```

Najbolji model

Iako su sva tri modela koja sam obradila pokazali poprilično slične rezultate, sa nekim odstupanjima, kao najbolji model izdvojila bih kNN jer je pokazao poboljšanje nakon oversamplinga gdje je dobio tačnost od skoro 0.80, osjetljivost 0.88, preciznost od 0.856 i kappa 0.42. Iako izvorno nije pokazao dobre rezultate, u ovom primjeru vidimo kako metoda poput oversamplinga može znatno promijeniti model.

Model logističke regresije je inicijalno pokazao dobar rezultat tačnosti i osjetljivosti, ali druge karakteristike nisu bile pretjernao zadovoljavajuće, tako da je ipak kNN klasificiran kao najbolji

model od trenutno obrađenih. Kako je obrađena samo ⅓ zadaće ostala dva modela nisu uzeta u razmatranje, a mislim da SVM i neuralne mreže mogu pokazati vrlo dobre rezultate, ali ovom analizom se ovako pokazalo.