Homework1

Paolo Fabbri - ID: 257552

2025-03-25

1. Introduction

In questa analisi, esaminiamo l'effetto di diverse variabili cliniche e demografiche sul rischio di sviluppare la malattia coronarica (CHD) entro un periodo di dieci anni. Per raggiungere questo obiettivo, utilizziamo due approcci statistici distinti: la regressione logistica, che consente di stimare la probabilità di sviluppare CHD in funzione delle variabili predittive, e il classificatore k-NN, che basa le sue previsioni sulla similarità tra osservazioni. Attraverso il confronto delle prestazioni dei due modelli, valutiamo la loro capacità discriminativa e l'affidabilità delle stime, discutendo i limiti e le implicazioni dei risultati nel contesto epidemiologico.

1.1 Data Exploration

```
library(caret)
library(class)
library(ggplot2)
CHD <- read.csv("chd.csv", header = TRUE, sep = ",")

head(CHD)
```

```
     sex age education smoker cpd stroke HTN diabetes chol DBP   BMI HR CHD
1   Male  39         4      0   0      0   0        0  195  70 26.97 80  No
2 Female  46         2      0   0      0   0        0  250  81 28.73 95  No
3   Male  48         1      1  20      0   0        0  245  80 25.34 75  No
4 Female  61         3      1  30      0   1        0  225  95 28.58 65 Yes
5 Female  46         3      1  23      0   0        0  285  84 23.10 85  No
6 Female  43         2      0   0      0   1        0  228 110 30.30 77  No
```

```
summary(CHD)
```

```
     sex                  age          education         smoker
 Length:4238        Min.   :32.00   Min.   :1.000   Min.   :0.0000
 Class :character   1st Qu.:42.00   1st Qu.:1.000   1st Qu.:0.0000
 Mode  :character   Median :49.00   Median :2.000   Median :0.0000
                    Mean   :49.58   Mean   :1.979   Mean   :0.4941
                    3rd Qu.:56.00   3rd Qu.:3.000   3rd Qu.:1.0000
                    Max.   :70.00   Max.   :4.000   Max.   :1.0000
                                    NA's   :105
      cpd              stroke              HTN             diabetes
 Min.   : 0.000   Min.   :0.000000   Min.   :0.0000   Min.   :0.00000
 1st Qu.: 0.000   1st Qu.:0.000000   1st Qu.:0.0000   1st Qu.:0.00000
 Median : 0.000   Median :0.000000   Median :0.0000   Median :0.00000
 Mean   : 9.003   Mean   :0.005899   Mean   :0.3105   Mean   :0.02572
 3rd Qu.:20.000   3rd Qu.:0.000000   3rd Qu.:1.0000   3rd Qu.:0.00000
 Max.   :70.000   Max.   :1.000000   Max.   :1.0000   Max.   :1.00000
 NA's   :29
      chol            DBP              BMI              HR
 Min.   :107.0   Min.   : 48.00   Min.   :15.54   Min.   : 44.00
 1st Qu.:206.0   1st Qu.: 75.00   1st Qu.:23.07   1st Qu.: 68.00
 Median :234.0   Median : 82.00   Median :25.40   Median : 75.00
 Mean   :236.7   Mean   : 82.89   Mean   :25.80   Mean   : 75.88
 3rd Qu.:263.0   3rd Qu.: 89.88   3rd Qu.:28.04   3rd Qu.: 83.00
 Max.   :696.0   Max.   :142.50   Max.   :56.80   Max.   :143.00
 NA's   :50                       NA's   :19      NA's   :1
     CHD
 Length:4238
 Class :character
 Mode  :character
```

```
str(CHD)
```

```
'data.frame':   4238 obs. of  13 variables:
 $ sex      : chr  "Male" "Female" "Male" "Female" ...
 $ age      : int  39 46 48 61 46 43 63 45 52 43 ...
 $ education: int  4 2 1 3 3 2 1 2 1 1 ...
```

```
 $ smoker   : int  0 0 1 1 1 0 0 1 0 1 ...
 $ cpd      : int  0 0 20 30 23 0 0 20 0 30 ...
 $ stroke   : int  0 0 0 0 0 0 0 0 0 0 ...
 $ HTN      : int  0 0 0 1 0 1 0 0 1 1 ...
 $ diabetes : int  0 0 0 0 0 0 0 0 0 0 ...
 $ chol     : int  195 250 245 225 285 228 205 313 260 225 ...
 $ DBP      : num  70 81 80 95 84 110 71 71 89 107 ...
 $ BMI      : num  27 28.7 25.3 28.6 23.1 ...
 $ HR       : int  80 95 75 65 85 77 60 79 76 93 ...
 $ CHD      : chr  "No" "No" "No" "Yes" ...
```

```r
#The output shows that some variables have missing values (NA):
#- Education
#- Cpd
#- Chol
#- BMI
#- HR
colMeans(is.na(CHD[, c("education","cpd", "chol", "BMI", "HR")])) * 100
```

```
 education        cpd       chol        BMI         HR
2.47758377 0.68428504 1.17980179 0.44832468 0.02359604
```

```r
#The percentage of missing values for these variables is very low, none of them exceeds 5%, s

#Factorizing categorical variables
CHD$sex <- factor(CHD$sex, labels = c("Female", "Male"))
CHD$CHD <- factor(CHD$CHD, labels = c("No", "Yes"))
CHD$education <- factor(CHD$education, labels = c("Low", "Medium-Low", "Medium-High", "High")
CHD$smoker <- factor(CHD$smoker, labels = c("No", "Yes"))
CHD$stroke <- factor(CHD$stroke, labels = c("No", "Yes"))
CHD$HTN <- factor(CHD$HTN, labels = c("No", "Yes"))
CHD$diabetes <- factor(CHD$diabetes, labels = c("No", "Yes"))

#Removing rows with NA values from the dataset
CHD <- na.omit(CHD)
summary(CHD)
```

```
     sex            age              education    smoker          cpd
 Female:2297   Min.   :32.00   Low        :1681   No :2059   Min.   : 0.00
 Male  :1742   1st Qu.:42.00   Medium-Low :1220   Yes:1980   1st Qu.: 0.00
               Median :49.00   Medium-High: 673              Median : 0.00
```

```
                Mean   :49.53   High       : 465              Mean   : 9.01
                3rd Qu.:56.00                                  3rd Qu.:20.00
                Max.   :70.00                                  Max.   :70.00
 stroke      HTN       diabetes        chol            DBP
 No :4016   No :2783   No :3936   Min.   :113.0   Min.   : 48.00
 Yes:  23   Yes:1256   Yes: 103   1st Qu.:206.0   1st Qu.: 75.00
                                  Median :234.0   Median : 82.00
                                  Mean   :236.7   Mean   : 82.87
                                  3rd Qu.:263.0   3rd Qu.: 89.50
                                  Max.   :600.0   Max.   :142.50
      BMI              HR          CHD
 Min.   :15.54   Min.   : 44.00   No :3433
 1st Qu.:23.05   1st Qu.: 68.00   Yes: 606
 Median :25.36   Median : 75.00
 Mean   :25.77   Mean   : 75.87
 3rd Qu.:27.99   3rd Qu.: 83.00
 Max.   :56.80   Max.   :143.00
```

1.2 Discriminative power of the predictors

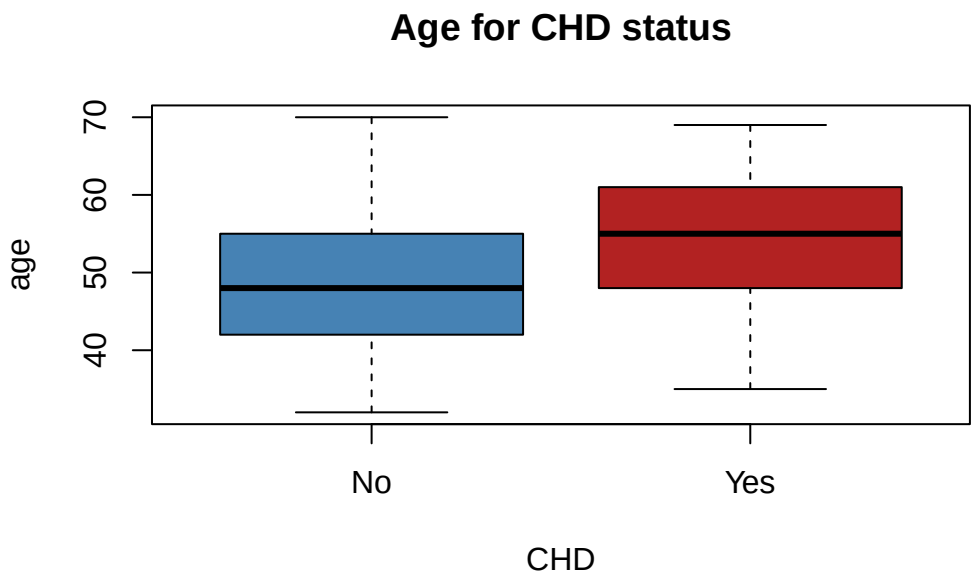
```r
table(CHD$CHD)
```

```
  No  Yes
3433  606
```

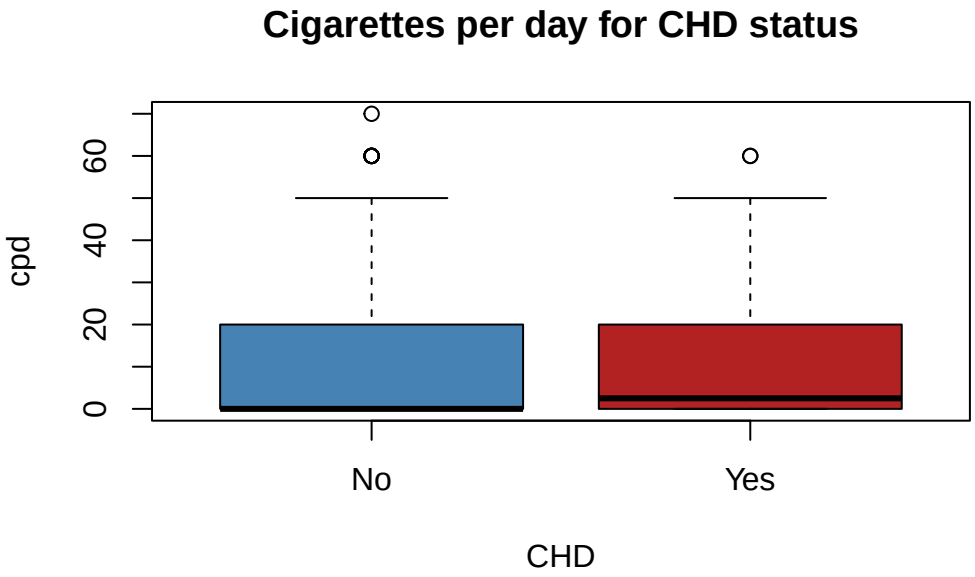
```r
prop.table(table(CHD$CHD))
```

```
       No       Yes
0.8499629 0.1500371
```

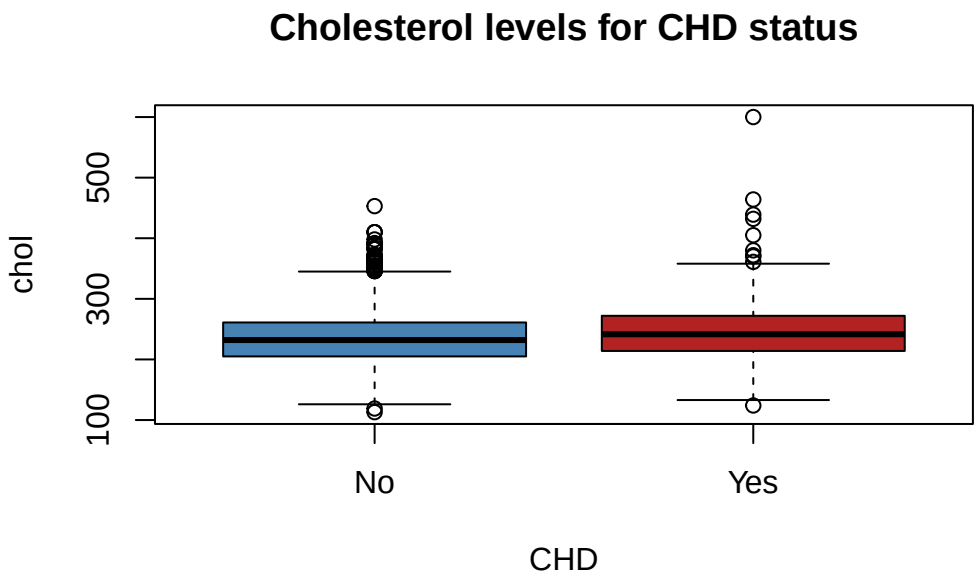
```r
#Visualization of the discriminative power for continuous variables
boxplot(age ~ CHD, data = CHD, main = "Age for CHD status", col = c("steelblue", "firebrick"))
```



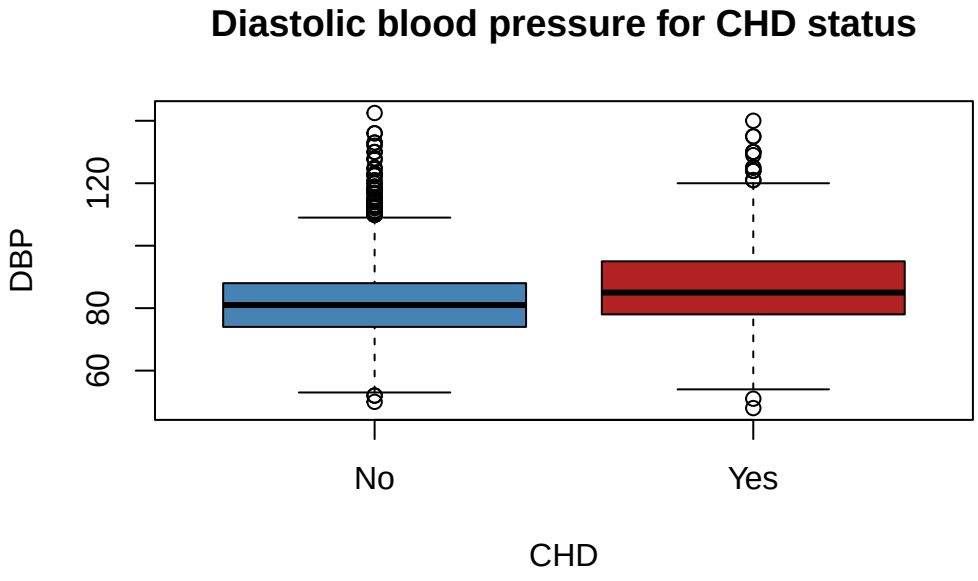

```
boxplot(cpd ~ CHD, data = CHD, main = "Cigarettes per day for CHD status", col = c("steelblue
```



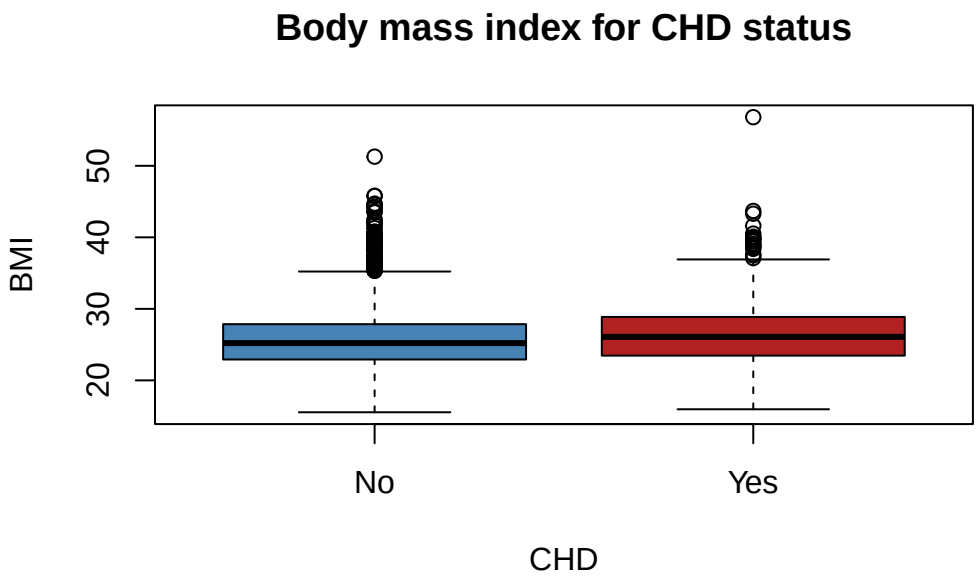
```r
boxplot(chol ~ CHD, data = CHD, main = "Cholesterol levels for CHD status", col = c("steelblu
```



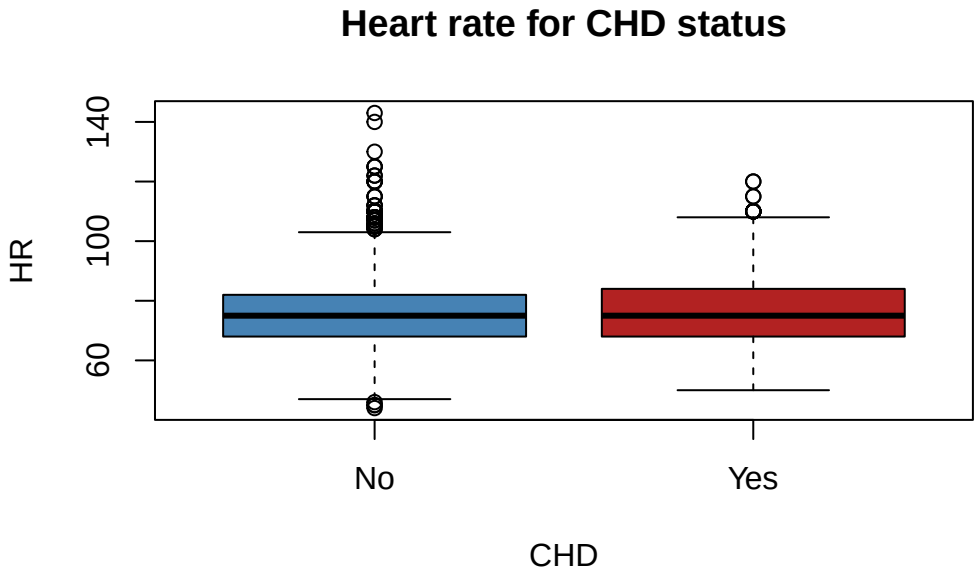

```r
boxplot(DBP ~ CHD, data = CHD, main = "Diastolic blood pressure for CHD status", col = c("ste
```



```r
boxplot(BMI ~ CHD, data = CHD, main = "Body mass index for CHD status", col = c("steelblue",
```



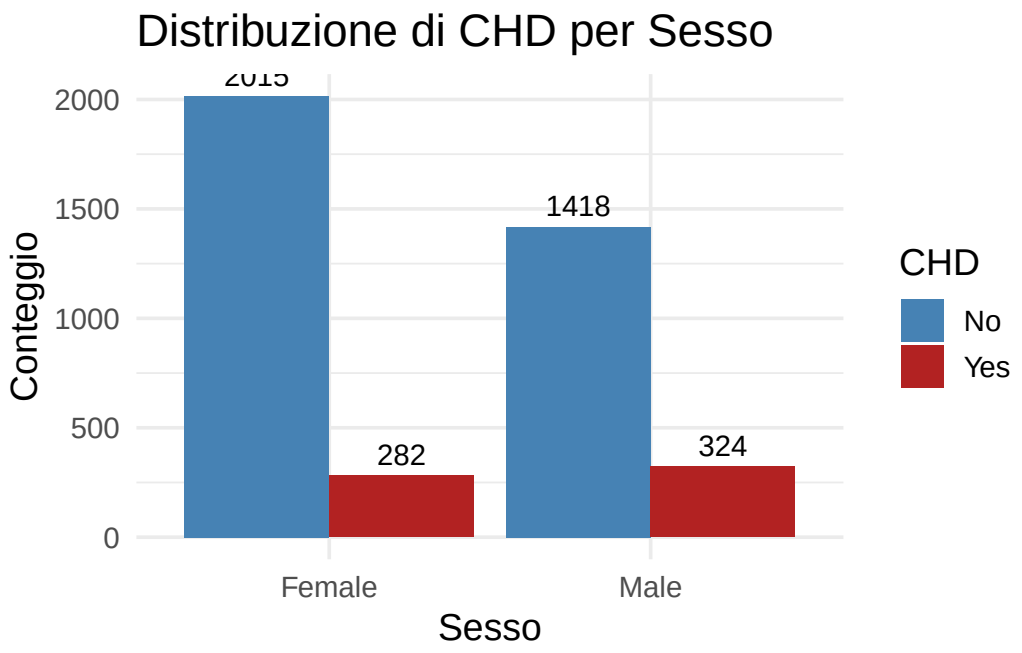

```r
boxplot(HR ~ CHD, data = CHD, main = "Heart rate for CHD status", col = c("steelblue", "firel
```



```r
#Visualization of the discriminative power for categorical variables
table(CHD$sex, CHD$CHD)
```

```
          No  Yes
  Female 2015  282
  Male   1418  324
```

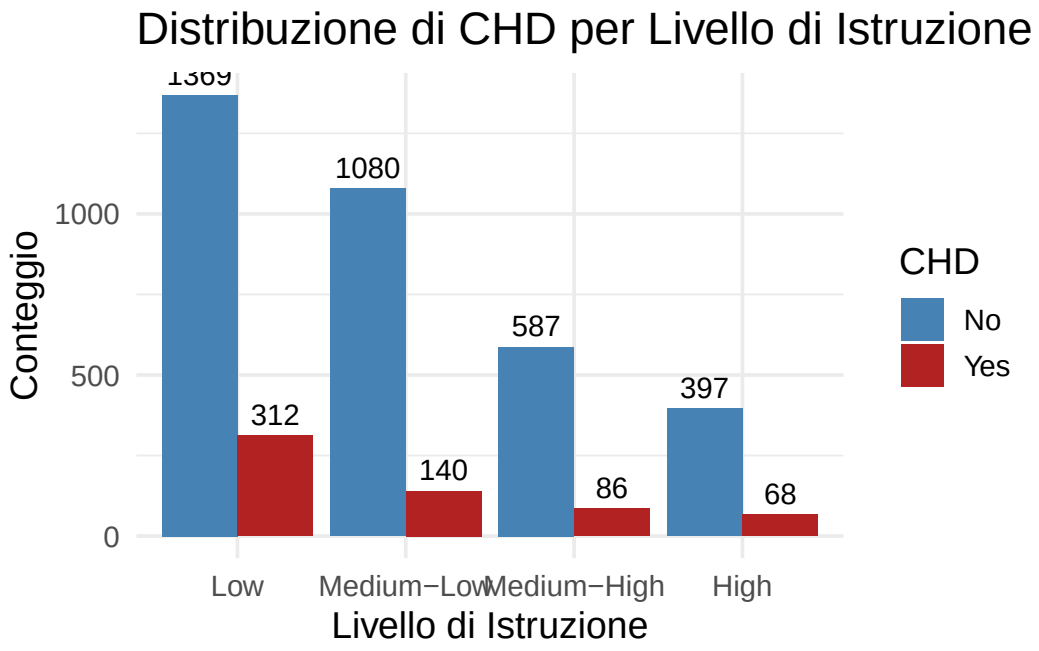
```r
ggplot(CHD, aes(x = sex, fill = CHD)) +
  geom_bar(position = "dodge") +
  geom_text(stat = "count", aes(label = after_stat(count)),
            position = position_dodge(width = 0.9), vjust = -0.5) +
  labs(title = "Distribuzione di CHD per Sesso",
       x = "Sesso", y = "Conteggio", fill = "CHD") +
  scale_fill_manual(values = c("steelblue", "firebrick")) +
  theme_minimal(base_size = 14)
```




```r
#Men seem to have a higher number of CHD cases compared to women, despite there being fewer r


table(CHD$education, CHD$CHD)
```

```
             No  Yes
Low        1369  312
Medium-Low 1080  140
Medium-High  587   86
High         397   68
```

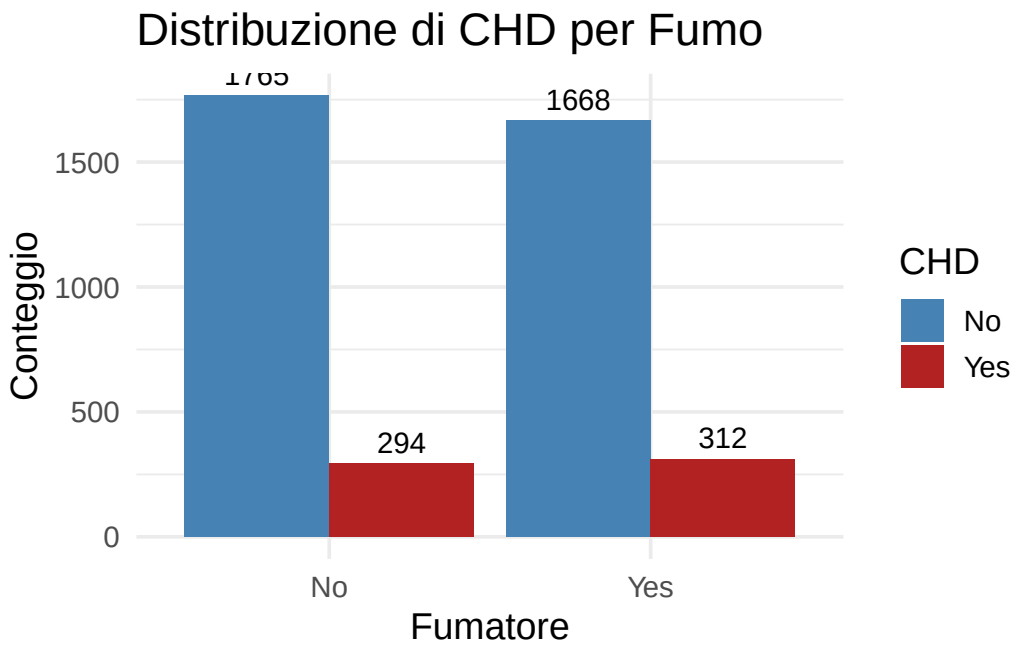
```
ggplot(CHD, aes(x = education, fill = CHD)) +
  geom_bar(position = "dodge") +
  geom_text(stat = "count", aes(label = after_stat(count)),
            position = position_dodge(width = 0.9), vjust = -0.5) +
  labs(title = "Distribuzione di CHD per Livello di Istruzione",
       x = "Livello di Istruzione", y = "Conteggio", fill = "CHD") +
  scale_fill_manual(values = c("steelblue", "firebrick")) +
  theme_minimal(base_size = 14)
```




```
#People with the lowest education level have the highest number of CHD cases, suggesting that

table(CHD$smoker, CHD$CHD)
```

```
       No  Yes
  No  1765  294
  Yes 1668  312
```

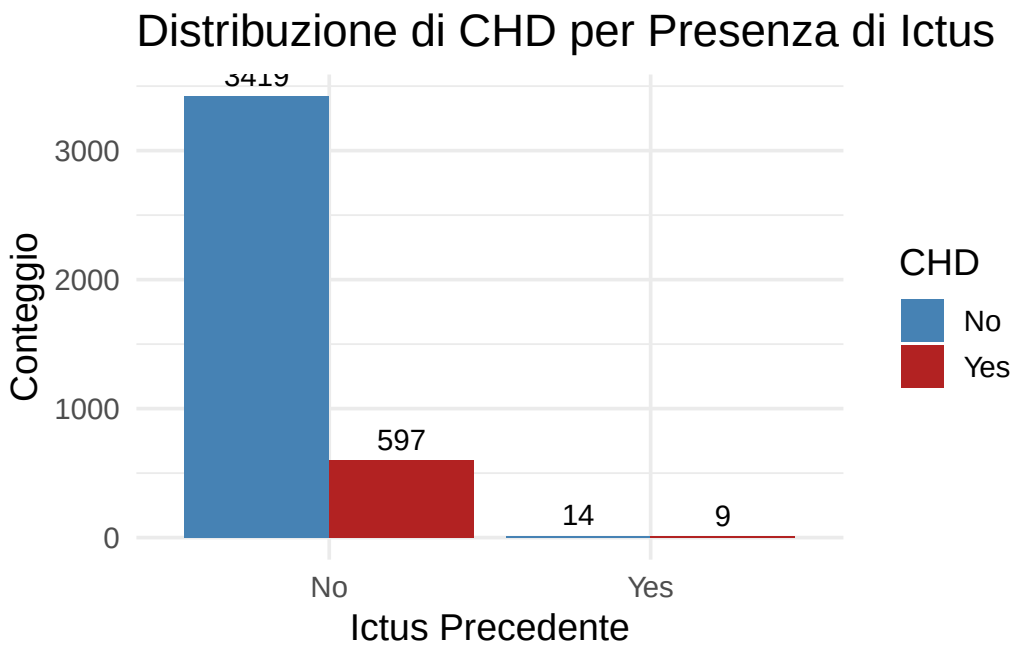
```
ggplot(CHD, aes(x = smoker, fill = CHD)) +
  geom_bar(position = "dodge") +
  geom_text(stat = "count", aes(label = after_stat(count)),
            position = position_dodge(width = 0.9), vjust = -0.5) +
  labs(title = "Distribuzione di CHD per Fumo",
       x = "Fumatore", y = "Conteggio", fill = "CHD") +
  scale_fill_manual(values = c("steelblue", "firebrick")) +
  theme_minimal(base_size = 14)
```




```
#The number of CHD cases between smokers and non-smokers (311) is very similar. Smoking does

table(CHD$stroke, CHD$CHD)
```

```
       No  Yes
  No  3419  597
  Yes   14    9
```

```
ggplot(CHD, aes(x = stroke, fill = CHD)) +
  geom_bar(position = "dodge") +
  geom_text(stat = "count", aes(label = after_stat(count)),
            position = position_dodge(width = 0.9), vjust = -0.5) +
  labs(title = "Distribuzione di CHD per Presenza di Ictus",
       x = "Ictus Precedente", y = "Conteggio", fill = "CHD") +
  scale_fill_manual(values = c("steelblue", "firebrick")) +
  theme_minimal(base_size = 14)
```

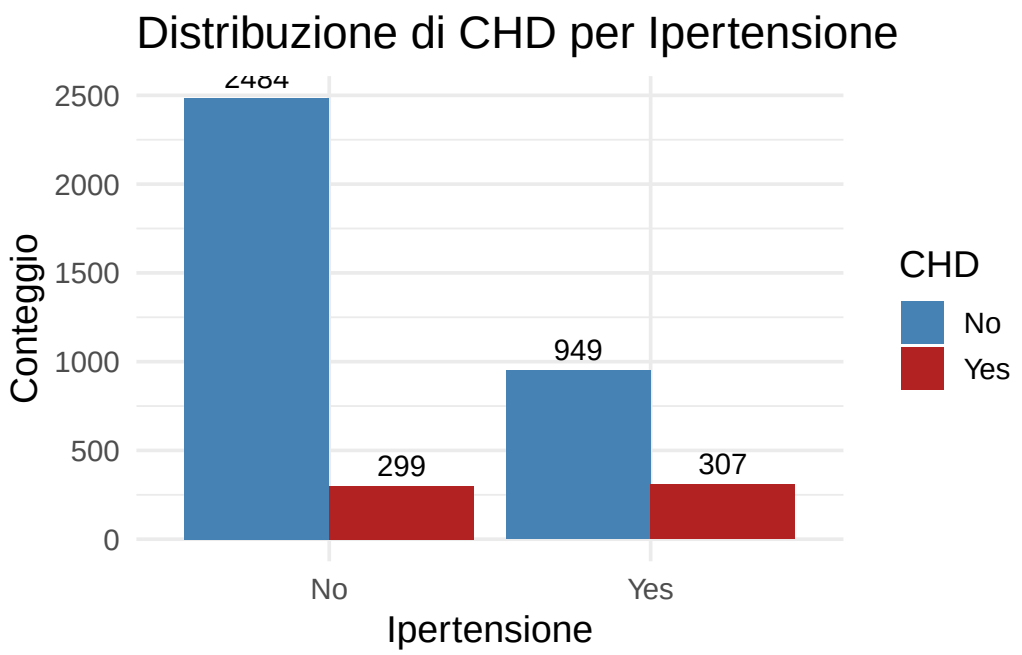



```
#Patients with a history of stroke (11 out of 25) show a high likelihood of developing CHD, l

table(CHD$HTN, CHD$CHD)
```

```
       No  Yes
  No  2484  299
  Yes  949  307
```

```
ggplot(CHD, aes(x = HTN, fill = CHD)) +
  geom_bar(position = "dodge") +
  geom_text(stat = "count", aes(label = after_stat(count)),
```

```
            position = position_dodge(width = 0.9), vjust = -0.5) +
  labs(title = "Distribuzione di CHD per Ipertensione",
       x = "Ipertensione", y = "Conteggio", fill = "CHD") +
  scale_fill_manual(values = c("steelblue", "firebrick")) +
  theme_minimal(base_size = 14)
```

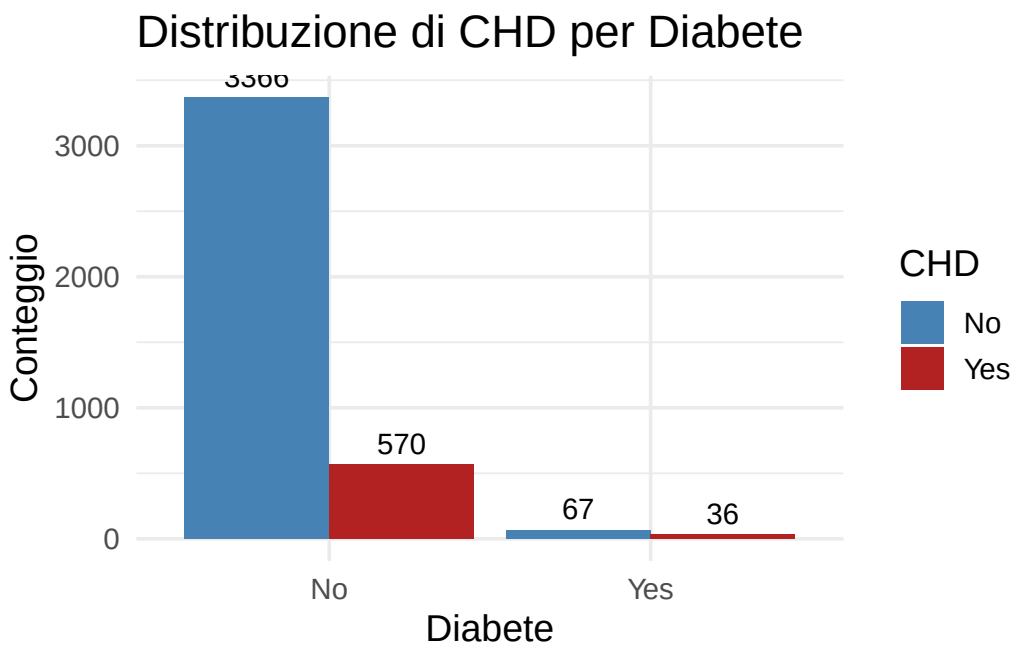



```
#Among individuals with hypertension, 325 out of 1316 developed CHD, compared to 319 out of 1

table(CHD$diabetes, CHD$CHD)
```

```
       No  Yes
  No  3366  570
  Yes   67   36
```

```
ggplot(CHD, aes(x = diabetes, fill = CHD)) +
  geom_bar(position = "dodge") +
  geom_text(stat = "count", aes(label = after_stat(count)),
            position = position_dodge(width = 0.9), vjust = -0.5) +
  labs(title = "Distribuzione di CHD per Diabete",
       x = "Diabete", y = "Conteggio", fill = "CHD") +
```

```
  scale_fill_manual(values = c("steelblue", "firebrick")) +
  theme_minimal(base_size = 14)
```




```
#Although the number of diabetic patients is small (109), the proportion of CHD among diabeti
```

2. Models

2.1 Logistic Regression Model

```
#Set seed for reproducibility
set.seed(123)

#Split data while maintaining the same CHD proportion
trainIndex <- createDataPartition(CHD$CHD, p = 0.8, list = FALSE)

#Create training set (80% of the data)
trainData <- CHD[trainIndex, ]

#Create test set (20% of the data)
```

```r
testData <- CHD[-trainIndex, ]
#Perform logistic regression on the training data to predict CHD and calculate test error
fit <- glm(CHD ~ ., data = trainData, family = "binomial")
summary(fit)
```

```
Call:
glm(formula = CHD ~ ., family = "binomial", data = trainData)

Coefficients:
                      Estimate Std. Error z value Pr(>|z|)
(Intercept)          -7.365348   0.737362  -9.989  < 2e-16 ***
sexMale               0.391028   0.114814   3.406  0.00066 ***
age                   0.072517   0.006977  10.394  < 2e-16 ***
educationMedium-Low  -0.135296   0.131323  -1.030  0.30289
educationMedium-High -0.022798   0.156550  -0.146  0.88422
educationHigh        -0.007878   0.175420  -0.045  0.96418
smokerYes            -0.079888   0.166574  -0.480  0.63151
cpd                   0.024907   0.006475   3.846  0.00012 ***
strokeYes             0.635669   0.514078   1.237  0.21626
HTNYes                0.422573   0.136469   3.096  0.00196 **
diabetesYes           0.974999   0.242097   4.027 5.64e-05 ***
chol                  0.001727   0.001192   1.448  0.14752
DBP                   0.015452   0.005385   2.869  0.00411 **
BMI                   0.002617   0.013432   0.195  0.84552
HR                   -0.006096   0.004514  -1.351  0.17682
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 2733.1  on 3231  degrees of freedom
Residual deviance: 2439.3  on 3217  degrees of freedom
AIC: 2469.3

Number of Fisher Scoring iterations: 5
```

```r
#Interpretation of results:
#Being male increases the risk of CHD (positive coefficient)
#Age has a positive effect on the probability of CHD (older individuals have higher risk)
#Cigarette consumption increases the risk of CHD
```

```r
#Hypertension increases the risk of CHD
#Diabetes strongly increases the likelihood of CHD
#Diastolic blood pressure (DBP) increases the risk of CHD

#Make predictions on the test data
lr.preds <- predict(fit, testData, type = "response")
#Set threshold for classification at 0.5
cl.preds <- ifelse(lr.preds >= 0.5, "Yes", "No")

#Confusion matrix
confusionMatrix(factor(cl.preds), testData$CHD)
```

```
Confusion Matrix and Statistics

          Reference
Prediction  No Yes
       No  683 117
       Yes   3   4

               Accuracy : 0.8513
                 95% CI : (0.8249, 0.8751)
    No Information Rate : 0.8501
    P-Value [Acc > NIR] : 0.4849

                  Kappa : 0.0469

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.99563
            Specificity : 0.03306
         Pos Pred Value : 0.85375
         Neg Pred Value : 0.57143
             Prevalence : 0.85006
         Detection Rate : 0.84634
   Detection Prevalence : 0.99133
      Balanced Accuracy : 0.51434

       'Positive' Class : No
```

```
#Model correctly classifies 85% of the cases
#Sensitivity (True Negative Rate - TNR): 99.56% → The model is very good at identifying patie
#Specificity (True Positive Rate - TPR): 3.31% → The model performs poorly in detecting patie

#Test error
mean(cl.preds != testData$CHD)
```

```
[1] 0.1486989
```

```
#The model has high specificity (good at detecting non-CHD patients), but is nearly useless i
```

2.2 K-NN Model

```
#Set seed for reproducibility
set.seed(123)
trainDataKnn <- trainData
testDataKnn <- testData

#Transform categorical variables into numeric variables for KNN training data
trainDataKnn$sex <- as.numeric(trainDataKnn$sex)
trainDataKnn$education <- as.numeric(trainDataKnn$education)
trainDataKnn$smoker <- as.numeric(trainDataKnn$smoker)
trainDataKnn$stroke <- as.numeric(trainDataKnn$stroke)
trainDataKnn$HTN <- as.numeric(trainDataKnn$HTN)
trainDataKnn$diabetes <- as.numeric(trainDataKnn$diabetes)

#Transform categorical variables into numeric variables for KNN test data
testDataKnn$sex <- as.numeric(testDataKnn$sex)
testDataKnn$education <- as.numeric(testDataKnn$education)
testDataKnn$smoker <- as.numeric(testDataKnn$smoker)
testDataKnn$stroke <- as.numeric(testDataKnn$stroke)
testDataKnn$HTN <- as.numeric(testDataKnn$HTN)
testDataKnn$diabetes <- as.numeric(testDataKnn$diabetes)


#Arrays to store errors and accuracies for each k value
errors <- numeric()
accuracies <- numeric()
```

```r
#Range of k values to try
for (k in 1:30) {
  #Perform KNN predictions with the current k value
  knn.pred <- knn(trainDataKnn[, -13], testDataKnn[, -13], trainDataKnn[, 13], k = k)

  #Calculate classification error
  error_rate <- mean(knn.pred != testDataKnn$CHD)

  #Calculate accuracy
  accuracy <- 1 - error_rate

  #Store error and accuracy in arrays
  errors[k] <- error_rate
  accuracies[k] <- accuracy
}

#Find the k value with the lowest error rate
best_k <- which.min(errors)
best_k
```

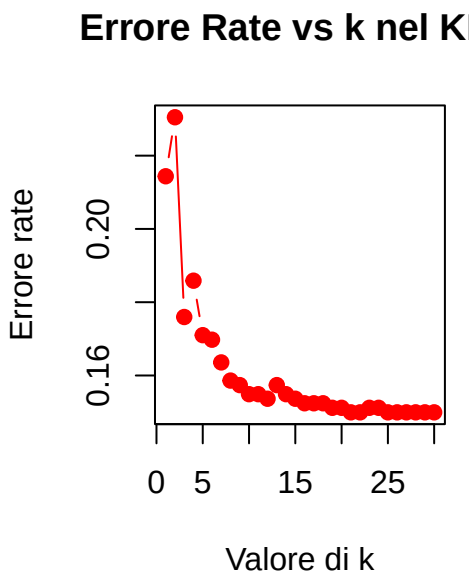
```
[1] 21
```

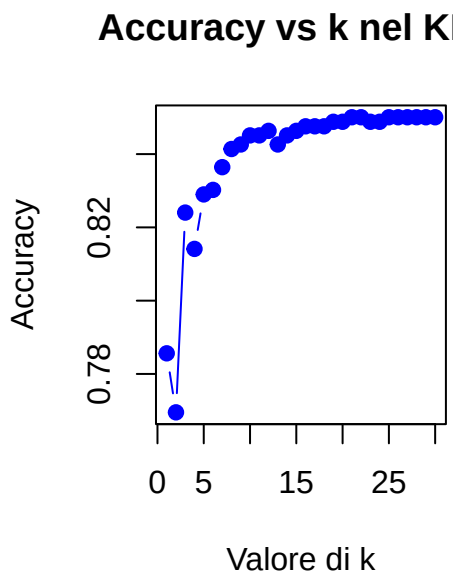
```r
knn.pred <- knn(trainDataKnn[, -13], testDataKnn[, -13], trainDataKnn[, 13], k = best_k)

#Confusion matrix
confusionMatrix(knn.pred, testDataKnn$CHD)
```

```
Confusion Matrix and Statistics

          Reference
Prediction  No Yes
       No  686 121
       Yes   0   0

               Accuracy : 0.8501
                 95% CI : (0.8235, 0.874)
    No Information Rate : 0.8501
    P-Value [Acc > NIR] : 0.5242

                  Kappa : 0
```

```
 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 1.0000
            Specificity : 0.0000
         Pos Pred Value : 0.8501
         Neg Pred Value :    NaN
             Prevalence : 0.8501
         Detection Rate : 0.8501
   Detection Prevalence : 1.0000
      Balanced Accuracy : 0.5000

       'Positive' Class : No
```

```r
#Interpretation:
#Accuracy: 0.8501, the model correctly predicts 85% of the cases
#Sensitivity: The model perfectly identifies patients with CHD
#Specificity: The model fails to recognize patients without CHD

# Despite the high accuracy (85%), the model is highly imbalanced. The low specificity indica


#Plot error rate vs. k
par(mfrow = c(1, 2))
plot(1:30, errors, type = "b", col = "red", pch = 19,
     xlab = "Valore di k", ylab = "Errore rate", main = "Errore Rate vs k nel KNN")

#Plot accuracy vs. k
plot(1:30, accuracies, type = "b", col = "blue", pch = 19,
     xlab = "Valore di k", ylab = "Accuracy", main = "Accuracy vs k nel KNN")
```





3. Conclusion

La scelta tra il modello **KNN** (K-Nearest Neighbors) e la **regressione logistica** dipende da vari fattori, tra cui le caratteristiche dei tuoi dati, la performance desiderata e la complessità del modello. Ecco una panoramica di entrambi i modelli per aiutarti a decidere:

1 Regressione Logistica:

Vantaggi:

-
- **Interpretabilità:** I coefficienti della regressione logistica possono essere interpretati in termini di probabilità e odds, il che aiuta a comprendere l'effetto delle variabili indipendenti sul risultato.
-
- **Semplicità:** È un modello relativamente semplice, facile da implementare e da capire.
-

- **Gestisce bene le variabili binarie:** Se stai cercando di prevedere un esito binario come nel caso del **CHD** (presenza o assenza), la regressione logistica è spesso un buon punto di partenza.
-
- **Probabilità:** La regressione logistica fornisce una stima delle probabilità di appartenere alla classe "Yes", il che può essere utile in applicazioni dove si vuole interpretare il rischio.
-

Svantaggi:

-
- **Assunzione di linearità:** La regressione logistica assume una relazione lineare tra le variabili indipendenti e la log-odds della variabile dipendente. Se i dati non seguono questa assunzione, il modello potrebbe non performare al meglio.
-
- **Problemi con il bilanciamento delle classi:** Se le classi sono molto sbilanciate (ad esempio, più pazienti senza CHD rispetto a quelli con CHD), la regressione logistica può soffrire in termini di accuratezza e performance.
-

2 KNN (K-Nearest Neighbors):

Vantaggi:

-
- **Non richiede assunzioni sulla forma dei dati:** KNN è un modello **non parametrico**, quindi non richiede che i dati siano distribuiti in un certo modo (ad esempio, non assume linearità).
-
- **Adatto per problemi complessi e non lineari:** KNN può essere molto utile se la relazione tra le variabili indipendenti e la variabile dipendente è complessa e non lineare.
-
- **Semplicità di implementazione:** Come la regressione logistica, KNN è relativamente semplice da implementare, anche se la scelta del parametro k e la normalizzazione dei dati possono essere cruciali per il suo successo.
-

Svantaggi:

-
- **Non interpretabilità:** KNN non fornisce una spiegazione esplicita del modello, quindi non è facile capire come le variabili influenzino il risultato. Non avrai coefficienti come nella regressione logistica.
-
- **Performance computazionale:** KNN può essere lento nei modelli con molti dati, poiché deve calcolare le distanze tra i punti ogni volta che fa una previsione.
-
- **Dipendenza dalla scelta di k:** La scelta di k è cruciale e può influenzare significativamente la performance del modello. Inoltre, il modello può essere influenzato da valori di k troppo piccoli o troppo grandi.
-

3 Combinazione di entrambi:

Potresti anche considerare di **combinare** i due modelli, utilizzando la **regressione logistica** come punto di partenza per interpretare le variabili e quindi usare **KNN** come un possibile modello complementare quando la relazione tra le variabili non è lineare.

Quale modello scegliere?

Sulla base dei tuoi risultati:

-
- **Regressione Logistica**: Ha mostrato una buona performance, ma i risultati di sensibilità e specificità potrebbero indicare che il modello sta soffrendo di **sbilanciamento delle classi**, dove ci sono più "No" rispetto a "Yes". La regressione logistica ti offre anche l'opportunità di interpretare l'effetto delle variabili sulle probabilità di sviluppare CHD.
-
- **KNN**: Ha mostrato una **alta accuracy (85%)**, ma con una **specificità pari a zero**, il che significa che non ha mai predetto correttamente "Yes" (chi ha il CHD). Questo potrebbe essere un forte svantaggio in un contesto sanitario, dove è cruciale non perdere casi positivi. La performance del modello potrebbe migliorare con la regolazione dei parametri (come il valore di k) o un bilanciamento delle classi.

-

Conclusione:

-
- **Se l'interpretabilità e la capacità di stimare le probabilità** sono importanti per il tuo caso d'uso, la **regressione logistica** è probabilmente la scelta migliore.
-
- **Se la precisione assoluta è l'obiettivo principale**, e sei disposto a esplorare ulteriormente il miglior valore di k o utilizzare tecniche di bilanciamento delle classi, **KNN** potrebbe essere un buon candidato, ma la **bassa specificità** è un punto critico che potrebbe richiedere un aggiustamento o un altro approccio.
-

Se hai bisogno di migliorare la performance del KNN, potresti considerare **tecniche di bilanciamento delle classi**, come la **sottocampionatura** o la **sovracampionatura** dei dati, per migliorare la capacità del modello di riconoscere i pazienti con CHD