Introduction to Data Science

Done By: Arpita Saha(19-41363-3)

Introduction: Data science is the field that applies information from data across a wide range of application fields by using scientific methods, procedures, algorithms, and systems to infer knowledge and insights from noisy, structured, and unstructured data. Data science is related to data mining, machine learning, big data, computational statistics and analytics. Data science is defined as a "concept that unifies statistics, data analysis, informatics, and their related approaches" in order to "understand and analyze actual phenomena" using data. In the context of mathematics, statistics, computer science, information science, and domain knowledge, it uses techniques and theories from several domains. Data science is distinct from computer science and information science. In short, we can say that A data scientist is someone who creates programming code and combines it with statistical knowledge to create insights from data.

Supervised learning is a sort of machine learning in which the output is predicted by the machines using well-labeled training data that has been used to train the machines. The term "labelled data" refers to input data that has already been assigned the appropriate output. In supervised learning, the training data that is given to the computers serve as the supervisor, instructing them on how to correctly predict the output.

Information about Dataset: To make a classification-based model we need a proper dataset.

Dataset Name: Maternal Health Risk Data Set Data Set

Link: https://archive.ics.uci.edu/ml/datasets/Maternal+Health+Risk+Data+Set

The dataset is based on a classification dataset which is based on real data which means this project will have a real-life impact. The dataset has 1014 instances and 7 attributes. From them 6 are important or can be called the feature matrix and the last one is the class attribute which is used for prediction. This dataset is about predicting the health risk of women during pregnancy based on some attributes. Now let us know about those parameters and why they are important for maternal health risk.

The class attribute is:

- **Risk level**

The other attributes are:

- **Age:** Any age in years when a woman during pregnant.
- **SystolicBP:** Our second attribute is SystolicBP which is given in mmHg. When a woman is pregnant always try to get an average BP it will avoid high health risk
- **DiastolicBP:** DiastolicBP is the lower value of blood pressure it's presented by mmHg.75-80mmhg is better for pregnant women. This range has low risk for health.
- **BS:** BS is same as important for women during pregnancy. If sugar level is more than 6.5 it's risky for a woman.
- **Body Temp:** Body temp is also very important for a pregnant woman**.**
- **HeartRate:** A normal resting heart rate in beats per minute. It's very important for pregnant women.

There is a total of 1014 instances of 7 attributes and all these instances were used for classification.

Task1: Import the dataset

```
dataset<- read.csv("D:/FALL 2022-23/INTRODUCTION TO DATA SCIENCE/PROJECT/Maternal Health Risk Data Set.csv",
                   header = TRUE,sep = ",")
dataset
```

Output:

dataset × final_project.R ×

Filter

	Age	SystolicBP	DiastolicBP	BS	BodyTemp	HeartRate	RiskLevel
1	25	130	80	15.00	98	86	high risk
2	35	140	90	13.00	98	70	high risk
3	29	90	70	8.00	100	80	high risk
4	30	140	85	7.00	98	70	high risk
5	35	120	60	6.10	98	76	low risk
6	23	140	80	7.01	98	70	high risk
7	23	130	70	7.01	98	78	mid risk
8	35	85	60	11.00	102	86	high risk
9	32	120	90	6.90	98	70	mid risk
10	42	130	80	18.00	98	70	high risk
11	23	90	60	7.01	98	76	low risk
12	19	120	80	7.00	98	70	mid risk
13	25	110	89	7.01	98	77	low risk
14	20	120	75	7.01	100	70	mid risk
15	48	120	80	11.00	98	88	mid risk
16	15	120	80	7.01	98	70	low risk
17	50	140	90	15.00	98	90	high risk
18	25	140	100	7.01	98	80	high risk
19	30	120	80	6.90	101	76	mid risk
20	10	70	50	6.90	98	70	low risk
21	40	140	100	18.00	98	90	high risk
22	50	140	80	6.70	98	70	mid risk

Task2: Details about dataset:

```
nrow(dataset)
ncol(dataset)
dim(dataset)
length(dataset)
names(dataset)
str(dataset)
```

Output:

```
> nrow(dataset)
[1] 1014
> ncol(dataset)
[1] 7
> dim(dataset)
[1] 1014    7
> length(dataset)
[1] 7
> names(dataset)
[1] "Age"         "SystolicBP"  "DiastolicBP" "BS"          "BodyTemp"    "HeartRate"   "RiskLevel"
> str(dataset)
'data.frame':   1014 obs. of  7 variables:
 $ Age        : int  25 35 29 30 35 23 23 35 32 42 ...
 $ SystolicBP : int  130 140 90 140 120 140 130 85 120 130 ...
 $ DiastolicBP: int  80 90 70 85 60 80 70 60 90 80 ...
 $ BS         : num  15 13 8 7 6.1 7.01 7.01 11 6.9 18 ...
 $ BodyTemp   : num  98 98 100 98 98 98 98 102 98 98 ...
 $ HeartRate  : int  86 70 80 70 76 70 78 86 70 70 ...
 $ RiskLevel  : chr  "high risk" "high risk" "high risk" "high risk" ...
>
```

Task3: Normalize the attributes:

```
normalize_data <- function(x) {
  ((x - min(x)) / (max(x) - min(x))) }
norm_data1 <- as.data.frame(lapply(dataset[,1:6], normalize_data))
norm_data1
```

Output:

dataset × final_project.R × norm_data1 ×

Filter

	Age	SystolicBP	DiastolicBP	BS	BodyTemp	HeartRate
1	0.25000000	0.66666667	0.60784314	0.692307692	0.0	0.9518072
2	0.41666667	0.77777778	0.80392157	0.538461538	0.0	0.7590361
3	0.31666667	0.22222222	0.41176471	0.153846154	0.4	0.8795181
4	0.33333333	0.77777778	0.70588235	0.076923077	0.0	0.7590361
5	0.41666667	0.55555556	0.21568627	0.007692308	0.0	0.8313253
6	0.21666667	0.77777778	0.60784314	0.077692308	0.0	0.7590361
7	0.21666667	0.66666667	0.41176471	0.077692308	0.0	0.8554217
8	0.41666667	0.16666667	0.21568627	0.384615385	0.8	0.9518072
9	0.36666667	0.55555556	0.80392157	0.069230769	0.0	0.7590361
10	0.53333333	0.66666667	0.60784314	0.923076923	0.0	0.7590361
11	0.21666667	0.22222222	0.21568627	0.077692308	0.0	0.8313253
12	0.15000000	0.55555556	0.60784314	0.076923077	0.0	0.7590361
13	0.25000000	0.44444444	0.78431373	0.077692308	0.0	0.8433735
14	0.16666667	0.55555556	0.50980392	0.077692308	0.4	0.7590361
15	0.63333333	0.55555556	0.60784314	0.384615385	0.0	0.9759036
16	0.08333333	0.55555556	0.60784314	0.077692308	0.0	0.7590361
17	0.66666667	0.77777778	0.80392157	0.692307692	0.0	1.0000000
18	0.25000000	0.77777778	1.00000000	0.077692308	0.0	0.8795181
19	0.33333333	0.55555556	0.60784314	0.069230769	0.6	0.8313253
20	0.00000000	0.00000000	0.01960784	0.069230769	0.0	0.7590361
21	0.50000000	0.77777778	1.00000000	0.923076923	0.0	1.0000000

Task4: Split the dataset into training set and test set

```
set.seed(123)

sample_set <- sample(c(TRUE, FALSE), nrow(norm_data1), replace=TRUE, prob=c(0.7,0.3))
train  <- norm_data1[sample_set, ]
test   <- norm_data1[!sample_set, ]

train_labels <- dataset[sample_set,7]


test_labels <-  dataset[!sample_set,7]
```

Output:

Trainset:

dataset × | final_project.R × | train × | norm_data1 ×

Filter

	Age	SystolicBP	DiastolicBP	BS	BodyTemp	HeartRate
1	0.25000000	0.66666667	0.60784314	0.692307692	0.0	0.9518072
3	0.31666667	0.22222222	0.41176471	0.153846154	0.4	0.8795181
6	0.21666667	0.77777778	0.60784314	0.077692308	0.0	0.7590361
7	0.21666667	0.66666667	0.41176471	0.077692308	0.0	0.8554217
9	0.36666667	0.55555556	0.80392157	0.069230769	0.0	0.7590361
10	0.53333333	0.66666667	0.60784314	0.923076923	0.0	0.7590361
12	0.15000000	0.55555556	0.60784314	0.076923077	0.0	0.7590361
13	0.25000000	0.44444444	0.78431373	0.077692308	0.0	0.8433735
14	0.16666667	0.55555556	0.50980392	0.077692308	0.4	0.7590361
15	0.63333333	0.55555556	0.60784314	0.384615385	0.0	0.9759036
17	0.66666667	0.77777778	0.80392157	0.692307692	0.0	1.0000000
18	0.25000000	0.77777778	1.00000000	0.077692308	0.0	0.8795181
19	0.33333333	0.55555556	0.60784314	0.069230769	0.6	0.8313253
22	0.66666667	0.77777778	0.60784314	0.053846154	0.0	0.7590361
23	0.18333333	0.22222222	0.31372549	0.115384615	0.0	0.8313253
25	0.18333333	0.55555556	0.60784314	0.115384615	0.0	0.8313253
27	0.15000000	0.55555556	0.50980392	0.092307692	0.0	0.7108434
28	0.20000000	0.33333333	0.31372549	0.092307692	0.0	0.7590361
29	0.65000000	0.55555556	0.80392157	0.092307692	0.0	0.8433735
30	0.30000000	0.22222222	0.21568627	0.092307692	0.0	0.9036145

Testset:

	Age	SystolicBP	DiastolicBP	BS	BodyTemp	HeartRate
2	0.41666667	0.77777778	0.80392157	0.538461538	0.0	0.7590361
4	0.33333333	0.77777778	0.70588235	0.076923077	0.0	0.7590361
5	0.41666667	0.55555556	0.21568627	0.007692308	0.0	0.8313253
8	0.41666667	0.16666667	0.21568627	0.384615385	0.8	0.9518072
11	0.21666667	0.22222222	0.21568627	0.077692308	0.0	0.8313253
16	0.08333333	0.55555556	0.60784314	0.077692308	0.0	0.7590361
20	0.00000000	0.00000000	0.01960784	0.069230769	0.0	0.7590361
21	0.50000000	0.77777778	1.00000000	0.923076923	0.0	1.0000000
24	0.13333333	0.22222222	0.21568627	0.115384615	0.0	0.7590361
26	0.10000000	0.33333333	0.41176471	0.092307692	0.0	0.8795181
31	0.16666667	0.33333333	0.80392157	0.084615385	0.0	0.9759036
32	0.21666667	0.33333333	0.70588235	0.084615385	0.0	0.7108434
34	0.18333333	0.55555556	0.60784314	0.084615385	0.0	0.8433735
37	0.83333333	0.55555556	0.60784314	0.007692308	0.0	0.8192771
50	0.08333333	0.55555556	0.60784314	0.076923077	0.0	0.7590361
53	0.41666667	0.33333333	0.41176471	0.076923077	0.0	0.6385542
58	0.53333333	0.55555556	0.60784314	0.030769231	0.0	0.7590361
59	0.21666667	0.22222222	0.21568627	0.030769231	0.0	0.8313253
65	0.35000000	0.55555556	0.21568627	0.007692308	0.0	0.8313253
67	0.11666667	0.16666667	0.21568627	0.230769231	0.8	0.9518072

Here, dataset is split into 70% and 30%. The dataset is split 70% into training set and 30% into testset.

Task5: Building a model (KNN)

```
install.packages("class")
library(class)

length(train_labels)

model<-knn(train=train,test=test,cl=train_labels,k=27)
model
```

Output:

```
R 4.2.1 · ~/
> library(class)
Warning message:
package 'class' was built under R version 4.2.2
> length(train_labels)
[1] 716
> model<-knn(train=train,test=test,cl=train_labels,k=27)
> model
  [1] high risk low risk  mid risk  mid risk  low risk  mid risk  low risk  high risk low risk  low risk  low risk  low risk
 [13] mid risk  low risk  mid risk  low risk  low risk  low risk  mid risk  mid risk  mid risk  mid risk  mid risk  mid risk
 [25] low risk  mid risk  low risk  mid risk  mid risk  high risk mid risk  high risk high risk high risk high risk high risk
 [37] high risk high risk high risk high risk high risk high risk mid risk  high risk low risk  high risk low risk  low risk
 [49] high risk high risk high risk mid risk  mid risk  mid risk  low risk  low risk  low risk  low risk  mid risk  low risk
 [61] mid risk  low risk  mid risk  mid risk  mid risk  high risk mid risk  high risk high risk low risk  mid risk  low risk
 [73] mid risk  high risk low risk  high risk low risk  high risk low risk  mid risk  mid risk  mid risk  low risk  mid risk
 [85] low risk  high risk mid risk  low risk  mid risk  high risk high risk high risk low risk  low risk  high risk mid risk
 [97] low risk  mid risk  low risk  low risk  low risk  mid risk  high risk low risk  high risk low risk  low risk  low risk
[109] low risk  mid risk  low risk  mid risk  low risk  low risk  mid risk  low risk  low risk  mid risk  low risk  mid risk
[121] low risk  low risk  low risk  mid risk  mid risk  high risk low risk  mid risk  mid risk  high risk mid risk  high risk
[133] mid risk  high risk high risk mid risk  low risk  mid risk  high risk mid risk  mid risk  low risk  low risk  high risk
[145] mid risk  low risk  low risk  high risk low risk  low risk  high risk high risk low risk  mid risk  low risk  low risk
[157] low risk  low risk  mid risk  high risk high risk mid risk  low risk  mid risk  mid risk  mid risk  low risk  mid risk
[169] mid risk  low risk  mid risk  high risk high risk mid risk  high risk high risk low risk  low risk  mid risk  low risk
[181] low risk  mid risk  low risk  mid risk  low risk  low risk  high risk low risk  high risk low risk  high risk high risk
[193] high risk mid risk  low risk  mid risk  low risk  low risk  low risk  high risk mid risk  mid risk  low risk  low risk
[205] low risk  mid risk  high risk high risk high risk high risk high risk mid risk  low risk  mid risk  mid risk  mid risk
[217] low risk  low risk  mid risk  mid risk  mid risk  high risk mid risk  mid risk  mid risk  mid risk  mid risk  mid risk
[229] low risk  low risk  low risk  mid risk  mid risk  low risk  low risk  mid risk  low risk  low risk  mid risk  low risk
[241] mid risk  mid risk  mid risk  low risk  low risk  mid risk  high risk mid risk  low risk  low risk  low risk  low risk
[253] mid risk  low risk  high risk mid risk  mid risk  mid risk  low risk  mid risk  low risk  low risk  mid risk  mid risk
[265] high risk mid risk  low risk  mid risk  mid risk  low risk  low risk  mid risk  mid risk  mid risk  low risk  low risk
[277] mid risk  low risk  high risk low risk  mid risk  high risk low risk  high risk high risk high risk high risk mid risk
[289] high risk high risk mid risk  mid risk  high risk high risk high risk high risk mid risk  mid risk
Levels: high risk low risk mid risk
```

Task6: Accuracy rate of KNN model

```
ACC <- 100 * sum(test_labels == model)/NROW(test_labels)
ACC
```

Output:

```
> ACC <- 100 * sum(test_labels == model)/NROW(test_labels)
> ACC
[1] 63.08725
>
```

Task7: creating confusion matrix

```
table(model,test_labels)
```

Output:

```
> table(model,test_labels)
           test_labels
model        high risk low risk mid risk
  high risk         63        4        8
  low risk           6       74       31
  mid risk          17       44       51
> 
```

Here is the description about confusion matrix:
A confusion matrix is a table that is used to define the performance of a classification algorithm**.**

In this model the “**high risk**” is positive class and the negative class is “**low risk**” and “**mid risk**”

True positive (TP): A true positive is an outcome where the model correctly predicts the positive class. Here the value of true positive is 63 which means 63 instances are correctly predicted or classified by “high risk” class.

True negative (TN): A true negative is an outcome where the model correctly predicts the negative class. Here the value of the true negative is 193 predicted as negative and the outcome is also true.

False positive (FP): False positive means predicted as positive but the outcome is negative. The value of FP is 23 which was predicted as a high-risk class but it’s in low-risk class and mid-risk class.

False negative (FN): False negative is predicted as negative but the outcome is positive. So, here the value is 12 which was predicted in the low-risk class and mid-risk class but the outcome is in the high-risk class.