Labwork No.3. Classification of Diabetes.

In this tutorial, we consider a dataset Pima Indians Diabetes from the `mlbench` package. This data set contains descriptions of patients that are females at least 21 years old of Pima Indian heritage. The goal is to predict whether a patient has signs of diabetes. The data is supervised i.e. the labels are known.

```
library(mlbench)
data(PimaIndiansDiabetes2)
set.seed(0)
```

Question 1:

1. What type of features we have (numerical, categorical, rank)? How many missing values we have and which features contain them?
2. How many positive and negative examples we have? Is the data set balanced?

Question 2:

Using the following code, import the file `aux_functions.r` and convert the labels into the integer format:

```
source('aux_functions.r')
x = PimaIndiansDiabetes2[,-9]
y = PimaIndiansDiabetes2[,9]
y = negative_positive_class_labels(y)
```

1. Write a function that returns the accuracy score:

```
accuracy_score <- function(y_true, y_pred){

}
```

1. What is the accuracy score of a dummy classifier that predicts the negative class only?

Question 3:

In this question, we will study basic tricks to deal with missing values.

1. Firstly, replace the NAs by 0. Then, classify the data using the LDA from the `MASS` library in the leave-one-out fashion (see the documentation of `lda`) and report the performance.
2. Now, for each feature, replace the NAs by its mean value. Report the performance.
3. Finally, remove observations where the missing values appear. Report the performance and make the conclusion which imputation works better on this data set. Try to explain why this could happen. Save the imputed data with the best performance for further questions.

Question 4:

In this question, we compare various classification models by computing 10-fold cross-validation score.

1. Create a partition of the data set by 10 folds. Use the function `createFolds` from the library `caret`. Read the documentation to choose a convenient way to output folds.
2. Perform the 10-fold cross-validation, where the following models are compared: LDA, kNN (k=1), kNN (k=5), kNN (k=10), perceptron (eta=0.1, num_iter=1000). Then, report mean and standard deviation across 10 folds for each method. Comment observed results. The function `knn` can be taken from the library `class`.
3. What statistical test we can use to verify whether the classifier with the highest accuracy is significantly better than the others on some level of significance?

Question 5:

In this question, we will dive deeper to see whether classification results are adequate. For sake of simplicity, we will limit computations on one train/test by taking the 1st fold as the test test, whereas the rest folds for the train set.

1. Evaluate the confusion matrix for the classifier with the best accuracy score. Comment observed results and make a conclusion whether these results are appropriate for applications like diabet prediction.
2. Write a function that returns the sensitivity score a.k.a. the true positive rate. Would the sensitivity score be a more appropriate metric in our case?

```
sensitivity_score <- function(y_true, y_pred){

}
```

1. Compute the sensitivity score for each classification approach under consideration. Comment the observed results.