Assignment

In this assignment we will design simple classifiers for the IRIS dataset. recall that this dataset contains 3 flowers (Iris Versicolor, Iris Setosa and Iris Virginica) and each flower has 4 features: sepal-length, sepal-width, petal-length and petal-width The histogram of pairwise relationships is shown on the last

page. If we examine the histogram, we notice that it is easier to identify Setosa than the other two flowers. Therefore, we focus our analysis on just two flowers: Versicolor (label 0) and Virginica (label 1).

The IRIS dataset can be downloaded into a Pandas frame:

```
import numpy as np
import pandas as pdf

url = r"https://archive.ics.uci.edu/ml/" \
 + r"machine-learning-databases/iris/iris.data"
data = pd.read_csv(url, names=[
        "sepal-length", "sepal-width",
        "petal-length", "petal-width",
            "Class"])
```

In this assignment, we will consider two different approaches to classification:

1. Method I: construct very simple classifiers ("weak learners") based on single feature. Combine weak classifiers in an ensemble.

2. Method II: construct density distribution for each feature and design a "weak learner" based on density. Combine density-based classifiers in an ensemble.

For every classifier, you split data 50/50 into training and testing sets. You estimate parameters by using training set and compute performance measures by using the testing set.

Question 1:

1. download IRIS dataset, remove Setosa flowers and assign labels 0 to to Versicolor and 1 to Virginica.

2. for each label and feature compute statistical averages (from training set!) and put them in the following table:

Feature	μ_0	σ_0	μ_1	σ_1	μ_{all}	σ_{all}
Petal Lengh Petal Width Sepal Lengh Sepal Width						

Table 1: Statistical measurements

3. for each class, compute the correlation matrix for your 4 features. Which features have the highest and lowest correlations?

4. discuss your findings

Question 2:

1. generate histograms of pairwise relationships for a training set (include these histograms in submitted homework). X_rain. You can use "pairplot" method of the seaborn package:

```
import seaborn as sns

features = ["sepal-length", "sepal-width",
            "petal-length", "petal-width"]

pair_plot = sns.pairplot(X_train[features])
plt.show()
```

2. examine the histograms and for each feature design a simple classifier ("weak learner") for labels. Your classifier can only consist of simple comparison using that single feature. For example,

```
if petal_length > 10:
    label = 0
else:
    label = 1
```

You design 4 such classfiers, one for each of the 4 features. Apply each classifier to testing set, compute the confusion matrix and summarize them in a table below

Classifier	TP	TN	FP	FN	Accuracy
(1) Petal Lengh (2) Petal Width (3) Sepal Lengh (4) Sepal Width					

Table 2: Results for signle-feature "weak" learners

3. discuss your findings and rank your "weak" learners by accuracy (from most accurate to least accurate)

Question 3:

You construct ensemble classifiers by combining 3 "weak learners" together. For example, you take (1), (2) and (3). The output label is the majority decision. You will have 4 such ensemble classifiers - there are 4 distinct ways to choose 3 classifiers out of 4.

1. For each such ensemble classifier, split data into training and test. Apply your classifiers on testing data, compute confusion matrix and summarize the results in a table below (note that no training is done, we are just combining the "weak" learners).

Ensemble	TP	TN	FP	FN	Accuracy
(1),(2),(3) (1),(2),(4) (1),(3),(4) (2),(3),(4)					

Table 3: Result for ensembles of "weak" learners

2. discuss your findings and rank your ensembles learners by accuracy (from most accurate to least accurate

3. compare "weak learners" and ensemble results.

Question 4:

We will now design a density-based "weak learner". Assume that for each label each feature is distributed according to a normal distribution with mean μ and standard deviation σ. Split the data and use training data to compute (μ_0, σ_0) for class 0 and (μ_1, σ_1) for class 1. Given a flower in testing set with a feature value x, you compute $p_0(x)$ and $p_1(x)$ and pick up the label corresponding to the larger value.

```
from scipy.stats import norm

p_0 = norm.pdf((x - mu_0)/sigma_0)
p_1 = norm.pdf((x - mu_1)/sigma_1)
if p_0 >= p_1:
```

```
    label = 0
else:
    label = 1
```

1. you design 4 such density-based classfiers, one for each of the 4 features. For each classifier, compute the confusion matrix (from a testing set! as before) and summarize them in a table below

Classifier (density)	TP	TN	FP	FN	Accuracy
(1) Petal Lengh (2) Petal Width (3) Sepal Lengh (4) Sepal Width					

Table 4: Results for Density-based "weak" learners

2. discuss your findings and rank your density-based "weak" learners by accuracy (from most accurate to least accurate

Question 5:

You construct density-based ensemble classifiers by combining 3 density-based classifiers together. For example, you take (1), (2) and (3). The output label is the majority decision. You will have 4 such ensemble classifiers - there are 4 distinct ways to choose 3 classifiers out of 4. Split your data again 50/50 into training and testing

1. For each such ensemble classifier, compute confusion matrix (on testing data!) and summarize the results in a table below

Ensemble (density)	TP	TN	FP	FN	Accuracy
(1),(2),(3) (1),(2),(4) (1),(3),(4) (2),(3),(4)					

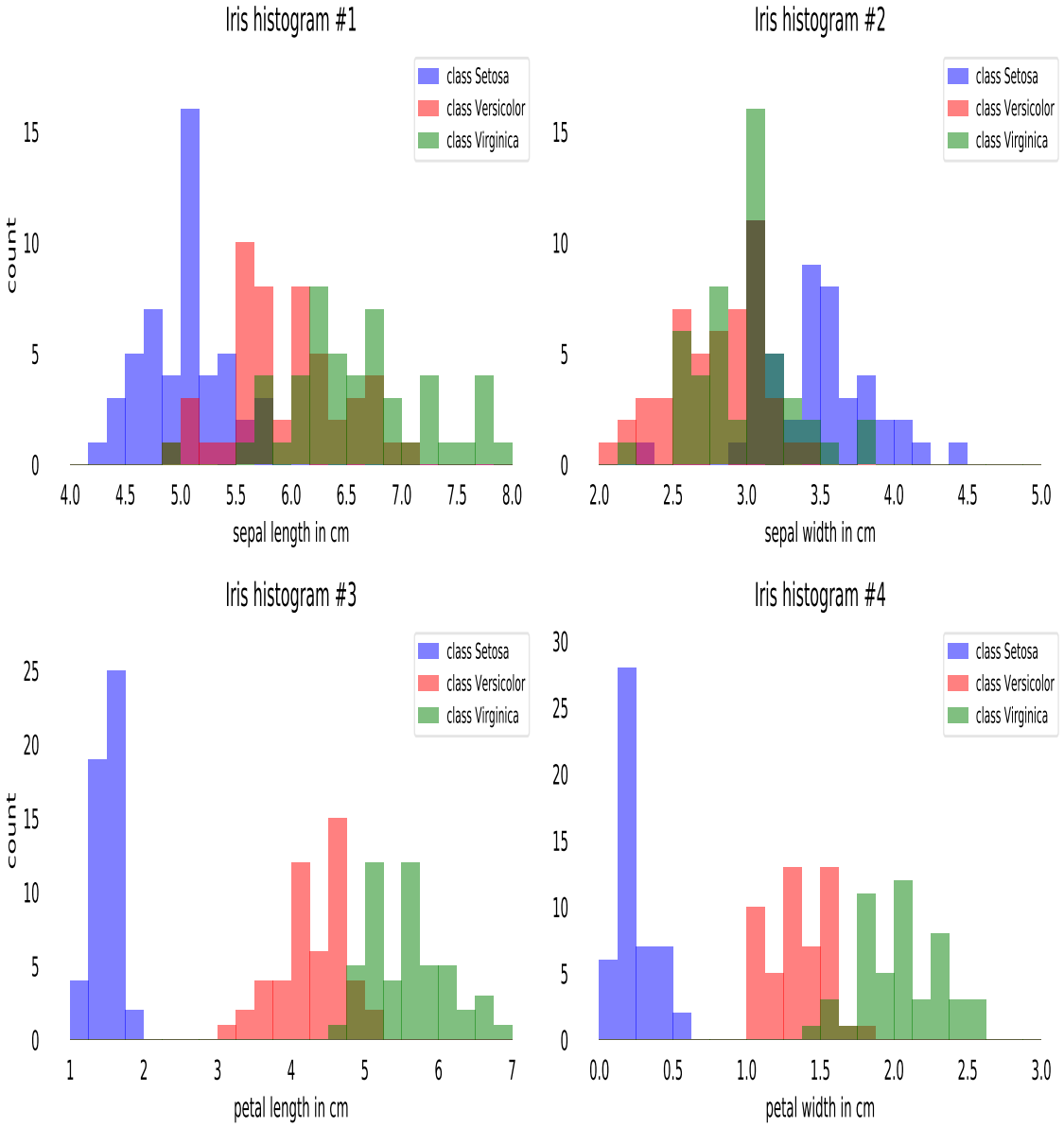
Table 5: Results for Density-based Ensembles

2. discuss your findings and rank your ensembles learners by accuracy (from most accurate to least accurate)

3. compare "weak learners" and ensemble results.

Question 6

1. give a quick summary on comparing classifiers in Method I and Method II


Iris histogram #1
class Setosa
class Versicolor
class Virginica
count
sepal length in cm
Iris histogram #2
sepal width in cm
Iris histogram #3
petal length in cm
Iris histogram #4
petal width in cm