**Технологии прикладного анализа данных**

**Лабораторная работа №3**

**Обучение без учителя**

Выполните следующие задания:

1. Загрузите датасет «Ирисы». Создайте два массива X и y, в которые запишите данные и целевую переменную соответственно.
2. Понизьте размерность данных при помощи следующего кода (не забудьте предварительно импортировать все необходимые библиотеки, например: from sklearn.decomposition import PCA):

pca = PCA(n\_components=2)

X\_reduced = pca.fit\_transform(X)

print("Meaning of the 2 components:")

**for** component **in** pca.components\_:

print(" + ".join("**%.3f** x **%s**" % (value, name)

**for** value, name **in** zip(component,

iris.feature\_names)))

plt.figure(figsize=(10,7))

plt.scatter(X\_reduced[:, 0], X\_reduced[:, 1], c=y, s=70, cmap='viridis')

plt.show()

1. Разбейте X и y, а также X\_reduced и y на тестовую и обучающие выборки. Для каждого из этих двух случаев постройте дерево решений DecisionTreeClassifier одной и той же глубины. Сравните результаты по метрике accuracy\_score.
2. Далее вы работаете с файлом nba\_2013.csv

player – name of the player

pos – the position of the player

g – number of games the player was in

pts – total points the player scored

fg. – field goal percentage

ft. – free throw percentage

1. Выполните обучение модели KMeans, подберите оптимальное количество кластеров (метрика accuracy\_score).
2. Визуализируйте оптимальное количество кластеров по двум признакам (один признак по оси x, другой по y, кластеры обозначены цветом точек).
3. Реализуйте иерархическую кластеризацию при помощи функции linkage (попробуйте различные методы и метрики).
4. Дважды сократите исходный набор данных, получив разные диаграммы.

Кратко опишите полученные результаты (ко всей работе).