




Практические задания по дисциплине «Системы и методы принятия решений»


Тема: «Метрические алгоритмы классификации»

Алгоритм ближайших соседей	
Баллы:	0 – 5.
Данные:	Ирисы Фишера (iris).
Язык программирования:	
Требования:	<ul style="list-style-type: none">• Реализовать алгоритм k ближайших соседей – kNN.• Выбрать оптимальное значение k по критерию скользящего контроля LOO и построить график зависимости $LOO(k)$.• Реализовать алгоритм k взвешенных ближайших соседей – $kwNN$.• Выбрать оптимальные значения k и q (параметра весовой функции q^j) по критерию скользящего контроля LOO и построить график зависимости $LOO(k, h)$.• Сравнить качество алгоритмов kNN и $kwNN$.• Построить карту классификации для рассматриваемых методов.• Привести пример, показывающий преимущество метода $kwNN$ над kNN.


Метод парзенковского окна	
Баллы:	0 – 5.
Данные:	Ирисы Фишера (iris).
Язык программирования:	
Требования:	<ul style="list-style-type: none">• Реализовать метод парзенковского окна.• Рассмотреть несколько видов ядер, для каждого из них построить карту классификации, оценить качество, подобрать оптимальное значение ширины окна h по критерию скользящего контроля LOO.• Сравнить качество построенных алгоритмов между собой и с ранее реализованными методами.


Метод потенциальных функций	
Баллы:	0 – 5.
Данные:	Ирисы Фишера (iris).
Язык программирования:	
Требования:	<ul style="list-style-type: none">• Реализовать метод потенциальных функций с фиксированной шириной окна.• Сделать чертеж демонстрирующий особенность метода (например, выделить с помощью размера и яркости объекты с более высоким потенциалом).• Рассмотреть несколько видов ядер, для каждого из них построить карту классификации, оценить качество.


	<ul style="list-style-type: none"> Сравнить качество построенного алгоритма классификации с ранее реализованными методами.
--	---


Отбор эталонных объектов	
Баллы:	0 – 10.
Данные:	Ирисы Фишера (iris).
Язык программирования:	
Требования:	<ul style="list-style-type: none"> Построить график отступов для объектов обучения относительно произвольного метрического классификатора. С помощью алгоритма «крутого склона» отобрать шумовые объекты – выбросы. Реализовать алгоритм STOLP для отбора опорных объектов. Сравнить качество и скорость работы алгоритма классификации до и после отбора объектов с помощью алгоритма STOLP.


Тема: «Байесовские алгоритмы классификации»

Линии уровня нормального распределения	
Баллы:	0 – 5.
Данные:	Центр и ковариационная матрица.
Язык программирования:	
Требования:	<ul style="list-style-type: none"> Отобразить на графике линии уровня нормального распределения с указанием на них значения плотности распределения. Рассмотреть все особые случаи.


Наивный нормальный байесовский классификатор	
Баллы:	0 – 5.
Данные:	Реальные или модельные.
Язык программирования:	
Требования:	<ul style="list-style-type: none"> Реализовать наивный нормальный байесовский классификатор. Сделать чертеж, демонстрирующий работу данного метода.


Подстановочный алгоритм (plug-in)	
Баллы:	0 – 5.
Данные:	Реальные или модельные.
Язык программирования:	
Требования:	<ul style="list-style-type: none"> Реализовать подстановочный байесовский алгоритм. Сделать чертеж, демонстрирующий работу алгоритма. Рассмотреть случаи, когда разделяющая кривая является: параболой, эллипсом и гиперболой.


Линейный дискриминант Фишера – ЛДФ	
Баллы:	0 – 5.
Данные:	Реальные или модельные.
Язык программирования:	
Требования:	<ul style="list-style-type: none"> • Реализовать линейный дискриминант Фишера. • Сделать чертеж, демонстрирующий работу алгоритма. • Указать на преимущества метода по сравнению с подстановочным алгоритмом.


Сеть радиальных базисных функций – RBF сеть	
Баллы:	0 – 20.
Данные:	Реальные или модельные, имеющие в каждом классе два и более сгустка.
Язык программирования:	
Требования:	<ul style="list-style-type: none"> • Реализовать EM-алгоритм с последовательным добавлением компонент. • на основе EM-алгоритма обучить RBF-сеть. • сделать чертеж, демонстрирующий работу алгоритма.

Тема «Линейные алгоритмы классификации»

ADALINE. Правило Хэбба (персептрон Розенблатта)	
Баллы:	0 – 5.
Данные:	Реальные или модельные.
Язык программирования:	
Требования:	<ul style="list-style-type: none"> • Реализовать метод стохастического градиента для произвольной функции потерь. • Реализовать линейный алгоритм классификации ADALINE. • Построить линейный классификатор с использованием правила Хэбба в методе стохастического градиента. • Сделать чертежи, демонстрирующие работу алгоритмов. • Сравнить два этих метода.

ADALINE. Правило Хэбба (персептрон Розенблатта)	
Баллы:	0 – 5.
Данные:	Реальные или модельные.
Язык программирования:	
Требования:	<ul style="list-style-type: none"> • Реализовать метод стохастического градиента для произвольной функции потерь. • Реализовать линейный алгоритм классификации ADALINE. • Построить линейный классификатор с использованием правила Хэбба в методе стохастического градиента. • Сделать чертежи, демонстрирующие работу алгоритмов. • Сравнить два этих метода.

Логистическая регрессия	
Баллы:	0 – 5.
Данные:	Реальные или модельные.
Язык программирования:	
Требования:	<ul style="list-style-type: none"> • Обучить алгоритм логистической регрессии с помощью метода стохастического градиента. • Сделать чертеж, демонстрирующий работу алгоритма. • В зависимости от значения апостериорной вероятности (принадлежности объекта к классу) раскрасить чертеж – чем большее значение вероятности, тем темнее цвет. • Продемонстрировать на одном графике работу реализованных выше алгоритмов.

Метод опорных векторов – SVM. ROC-кривая	
Баллы:	0 – 5.
Данные:	Реальные или модельные.
Язык программирования:	 + kernlab
Требования:	<ul style="list-style-type: none"> • С помощью библиотеки kernlab для языка R реализовать алгоритм SVM. • Реализовать алгоритм построения ROC-кривой и вычисления AUC. • Продемонстрировать работу метода для случаев: <ul style="list-style-type: none"> ○ линейно разделимой выборки; ○ линейно неразделимой выборки с малым числом объектов, «мешающих» линейной отделимости; ○ линейно неразделимой выборки с средним числом объектов, «мешающих» линейной отделимости.