МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

Федеральное государственное бюджетное образовательное

учреждение высшего образования

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»



Факультет прикладной математики и информатики

Кафедра ТПИ

Дисциплина: «Основы теории машинного обучения»

Лабораторная работа №1

Вариант №1

**ОЦЕНКА ПАРАМЕТРОВ И ПРОВЕРКА ГИПОТЕЗ В ЛИНЕЙНЫХ МОДЕЛЯХ С КАЧЕСТВЕННЫМИ ФАКТОРАМИ**

Факультет: ФПМИ

Группа: ПМИМ-31

Выполнили: Монгуш Н. С., Тарулин М. А., Филипенко Ю. Д.

Преподаватель: Попов А. А.

Дата выполнения:

Отметка о защите:

**Задание**

1. По данным варианта №1 сформировать матрицу наблюдений X, постулировать модель дисперсионного анализа с главными эффектами (без взаимодействия уровней факторов).
2. Провести редукцию модели к модели полного ранга, определить базис ФДО.
3. По методу МНК- оценивания провести оценивание ФДО в редуцированной модели. Проверить гипотезы о незначимости различий в эффектах уровней для каждого фактора и фактора в целом.

Таблица 1. Исходные данные

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Уровни фактора 1 | Уровни фактора 2 | | | |
| B1 | B2 | B3 | B4 |
| А1 | 3,1  2,9 | 4,1  3,9 | 2,1  1,9 | 2,1  1,9 |
| А2 | 3,9  4,1 | 4,9  5,1 | 2,9  3,1 | 3,0  3,0 |
| А3 | 2,1  1,9 | 3,1  2,9 | 1,1  0,9 | 1,0  1,0 |

**Ход работы**

1. Матрица наблюдений X будет иметь следующий вид:
2. Модель дисперсионного анализа с главными эффектами будет выглядеть следующим образом:

где – генеральное среднее, – эффекты первого и второго факторов соответственно, – ошибка эксперимента.

1. Для того, чтобы провести редуцирование модели к модели полного ранга можно воспользоваться факторизацией матрицы:

где – матрица полного строчного ранга:

матрица А задает вид базиса ФДО и имеет вид:

где – единичная матрица размера – числу линейно независимых столбцов X.

где – матрица с линейно зависимыми столбцами:

Базис ФДО для модели составляют функции:

1. Произведем МНК-оценивание ФДО редуцированной модели:

где – значения отклика, .

1. Проверка гипотез

Гипотезы о незначимости различий в эффектах уровней:

Гипотезы о незначимости фактора:

**Вывод**

В ходе анализа гипотез об отсутствии значимого влияния различных уровней, было обнаружено, что эффект от уровня является незначимым, тогда как факторы A и B обладают статистической значимостью.

**Код программы**

import sys

import numpy as np

import sympy as sp

from numpy.linalg import inv

sys.path.append('C:\\Users\\Mark\\PycharmProjects\\machineLearningNstu')

from src.data.varianсe\_analysis import get\_two\_factor\_observation\_matrix

def get\_two\_factor\_observation\_matrix(  
 num\_of\_factors\_1: int, num\_of\_factors\_2: int, observation\_numbers: int  
) -> np.ndarray | None:  
 observation\_matrix: np.ndarray = np.zeros(  
 (num\_of\_factors\_1\*num\_of\_factors\_2\*observation\_numbers, (num\_of\_factors\_1 + num\_of\_factors\_2 + 1)), dtype=int  
 )  
 observation\_matrix[:, 0] = 1  
  
 observation\_matrix[:, 1] = [number for number in [1, 0, 0] for \_ in range(num\_of\_factors\_2\*observation\_numbers)]  
 observation\_matrix[:, 2] = [number for number in [0, 1, 0] for \_ in range(num\_of\_factors\_2\*observation\_numbers)]  
 observation\_matrix[:, 3] = [number for number in [0, 0, 1] for \_ in range(num\_of\_factors\_2 \* observation\_numbers)]  
  
 observation\_matrix[:, 4] = [number for \_ in range(num\_of\_factors\_1) for number in [1, 1, 0, 0, 0, 0, 0, 0]]  
 observation\_matrix[:, 5] = [number for \_ in range(num\_of\_factors\_1) for number in [0, 0, 1, 1, 0, 0, 0, 0]]  
 observation\_matrix[:, 6] = [number for \_ in range(num\_of\_factors\_1) for number in [0, 0, 0, 0, 1, 1, 0, 0]]  
 observation\_matrix[:, 7] = [number for \_ in range(num\_of\_factors\_1) for number in [0, 0, 0, 0, 0, 0, 1, 1]]  
 return observation\_matrix

X: np.array = get\_two\_factor\_observation\_matrix(3, 4, 2)

X\_1: np.ndarray = np.delete(X, [3, -1], axis=1)

X\_2: np.ndarray = X[:, [3, -1]]

А\_wavy: np.ndarray = np.round(np.dot(np.dot(inv(np.dot(X\_1.T, X\_1)), X\_1.T), X\_2), 4)

I\_r: np.ndarray = np.eye(6)

A: np.ndarray = np.hstack([I\_r, А\_wavy])

X\_reduced: np.ndarray = np.round(np.dot(X\_1, A), 4)

Y = [

# A1

3.1, 2.9,

4.1, 3.9,

2.1, 1.9,

2.1, 1.9,

# A2

3.9, 4.1,

4.9, 5.1,

2.9, 3.1,

3.0, 3.0,

# A3

2.1, 1.9,

3.1, 2.9,

1.1, 0.9,

1.0, 1.0

]

Y: np.array = np.array([Y]).T

theta\_dashed: np.ndarray = np.round(np.dot(np.dot(inv(np.dot(X\_1.T, X\_1)), X\_1.T), Y), 4)