МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ОДЕСЬКА ПОЛІТЕХНІКА»
Навчально-науковий інститут комп'ютерних систем
Кафедра інформаційних систем

Денис СІДОРОВ
(група АІ-224)

ЛАБОРАТОРНА РОБОТА №4

З дисципліни «Методи та системи штучного інтелекту»

Моделювання задач класифікації засобами пакету Orange

Спеціальність:
F3 Комп'ютерні науки

Освітньо-професійна програма:
Комп'ютерні науки

Керівник:
Анатолій НІКОЛЕНКО, к.т.н., доцент

Одеса – 2025

Назва роботи. Моделювання задач класифікації засобами Orange.

Мета роботи. Навчитись реалізовувати алгоритми класифікації засобами та пакету Orange.

Завдання.

1. Використовуючи данні за варіантами, сформувати змістовну постановку задачі та детально описати вхідні дані.
2. Виконайте кроки 3-9 та дослідить можливості класифікаторів.
3. Проведіть серію експериментів з метою отримання кращих результатів класифікації шляхом зміни параметрів кожного з методів класифікації. Відобразіть всі вікна з параметрами віджетів.
4. Для нейронної мережі визначте мінімально необхідну кількість нейронів без втрати якості класифікації та функцію активації. Для методу k найближчих сусідів визначити найкраще значення k за метрикою F_1.
5. Відобразіть значення метрик із віджету Test and Score, а також структуру дерева рішень із віджету Tree Viewer.
6. Візьміть один об’єкт з вибірки і, згідно з структурою дерева рішень, покроково опишіть просування даного об’єкта від кореневого вузла до листа, в якому об’єкт буде класифікований. Необхідно дотримуватися умов переходу, що визначають по якому із ребер йти.
7. Підрахуйте помилки класифікації 1-го та 2-го роду та їх відносні частки істинно позитивних випадків та істинно негативних випадків, для кращого та для гіршого класифікаторів, згідно з метрикою F_1.
8. Відобразіть у протоколі графіки ROC-кривих, по одному графіку на клас, при цьому на кожному графіку відобразіть одночасно ROC-криві для всіх класифікаторів.
9. У верхньому правому куті вікна віджета Confusion Matrix є параметр Show, виберіть зі списку ліворуч кращий класифікатор за метрикою F_1 і для нього зробіть скріншот для протоколу при параметрі Show =

Number of Instances, другий скріншот при Show = Proportion of Actual. Для інших класифікаторів збережіть лише по одному скріншоту при Show = Proportion of Actual.

10. Оформити звіт з результати виконання кроків завдання.

Виконання роботи.

За умовою задачі, дана таблиця нерухомості (рис. 4.1). Необхідно, використовуючи оцінку (grade) як цільову змінну, провести класифікацію даних за допомогою нейронної мережі, логістичної регресії, методу k найближчих сусідів та дерева рішень.

df_train

date	price	bedrooms	grade	has_basement	living_in_m2	renovated	nice_view	perfect_condition	real_bathrooms	has_lavatory	single_floor	month	quartile_zone
2014-05-15	312000.0	2	2	True	138.42547	False	False	False	2	True	False	5	1
2014-11-14	310000.0	2	2	False	105.90942	False	False	False	1	True	False	11	3
2014-12-24	320000.0	2	2	False	117.98681	False	True	False	1	False	True	12	2
2015-02-22	264500.0	2	3	False	151.43189	False	False	False	2	True	True	2	1
2015-01-06	700000.0	3	2	True	341.88304	False	False	False	3	False	False	1	4
2015-03-04	445000.0	1	2	True	84.54173	False	True	False	1	False	True	3	3
2014-06-04	240500.0	2	2	True	135.63838	False	False	False	1	True	True	6	1
2014-10-22	235000.0	1	1	False	139.3545	False	False	False	1	False	True	10	1
2014-07-11	206000.0	2	2	False	105.90942	False	False	False	1	True	True	7	1
2014-09-03	545000.0	2	3	False	118.91584	False	False	False	2	True	False	9	4
2014-05-13	475000.0	3	5	True	347.45722	False	False	False	2	True	True	5	2
2014-10-02	456000.0	3	3	True	206.24466	False	False	False	1	True	True	10	4
2015-01-23	478000.0	2	2	True	145.85771	False	False	False	2	True	True	1	4
2014-07-12	690000.0	3	4	False	269.4187	False	False	False	2	True	False	7	4
2014-05-16	675000.0	3	4	False	297.2896	False	False	False	2	True	False	5	3
2014-07-30	442500.0	2	2	True	167.2254	False	False	False	1	True	True	7	2
2014-06-08	540000.0	3	3	True	266.63161	False	False	False	3	False	False	6	3
2014-07-22	255000.0	3	2	True	163.50928	False	False	False	2	True	True	7	1
2014-08-26	399950.0	2	2	True	136.56741	False	False	False	1	False	True	8	3
2014-11-17	455000.0	3	2	True	178.37376	False	False	False	1	True	True	11	2
2015-02-27	435000.0	1	2	True	91.97397	False	False	False	1	False	True	2	3
2014-10-29	470000.0	3	3	False	234.11556	False	False	False	2	True	False	10	2

Рисунок 4.1 – Фрагмент таблиці

В якості змінних для класифікації використаємо кількість спален, кількість ванних кімнат, площу у метрах квадратних та розподілення поштових індексів від дешевих до дорогих. Для вирішення задачі скористаємося засобами пакету Orange. Послідовність віджетів у середовищі

Orange для вирішення задачі на даних про нерухомість наведена на рисунку 4.2.

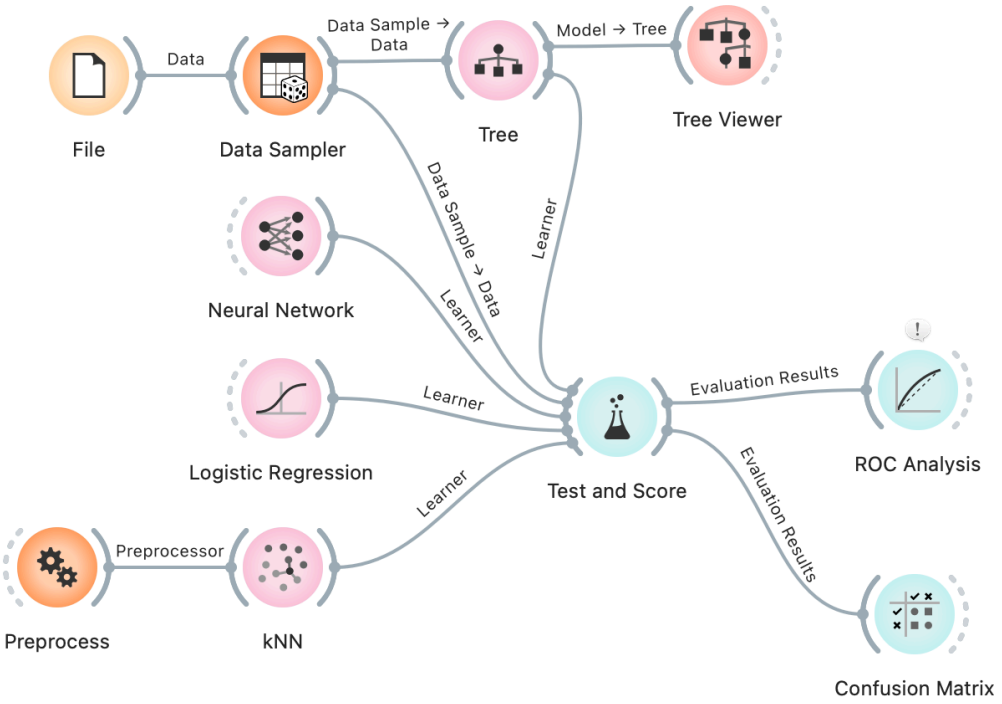
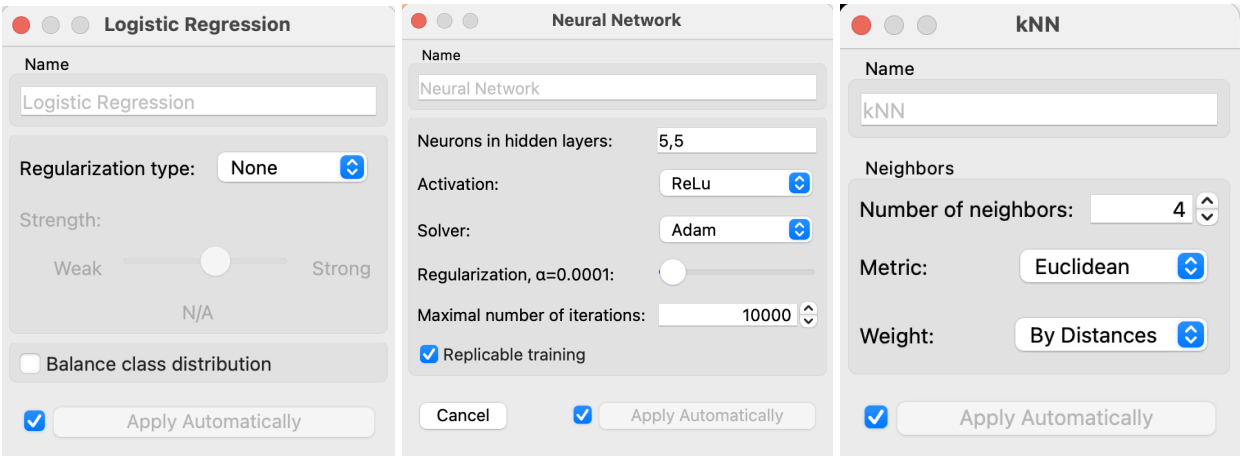


Рисунок 4.2 – Скріншот послідовності віджетів робочого процесу

З групи Model виберемо віджети Logistic Regression, Neural Network та kNN і налаштуємо відповідні параметри (рис. 4.3 а, б і в).



а б в

Рисунок 4.3 – Скріншоти параметрів віджетів Logistic Regression (а), Neural Network (б) та kNN (в)

Після проведенн експеримнтів, для нейронної мережі було встановлено оптимальне значення нейронів 5,5, а для методу k найближчих сусідів оптимальну кількість сусідів – 4. Саме ці значення дають найточніші результати.

До віджету kNN підключимо віджет Preprocess та встановимо нормалізацію значень ознак до інтервалу від 0 до 1 (рис. 4.4).

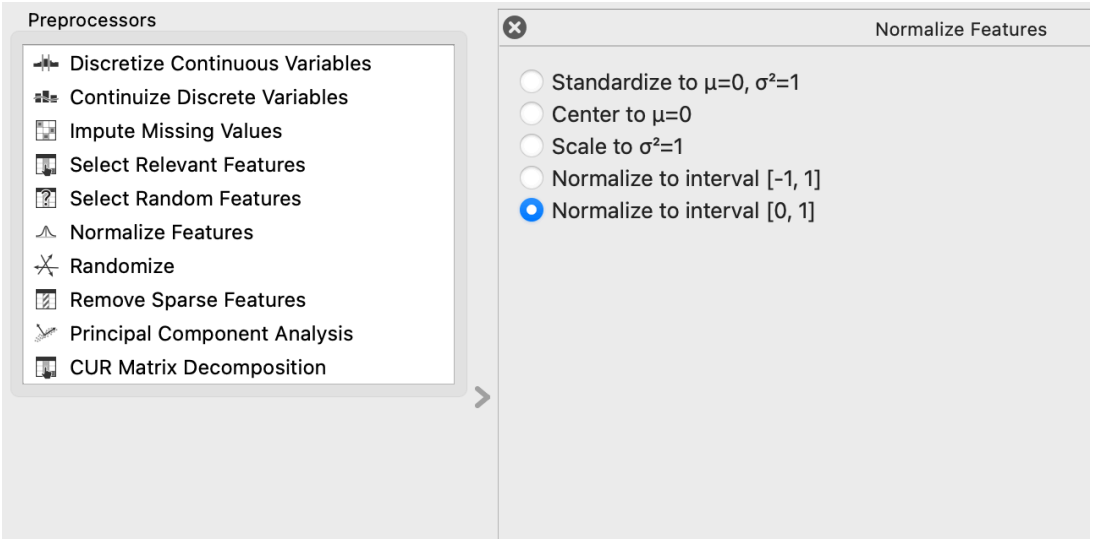


Рисунок 4.4 – Скріншот віджету Preprocess

Також додамо віджет Tree і налаштуємо його параметри (рис 4.5).

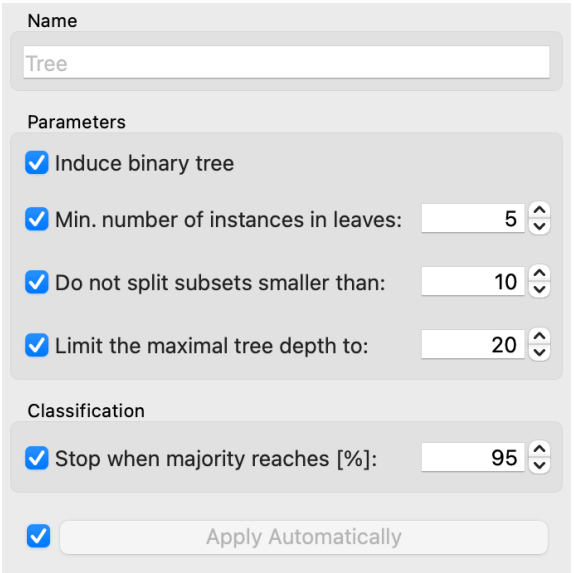


Рисунок 4.5 – Скріншот параметрів віджету Tree

У віджеті Tree Viewer побачимо результат роботи дерева рішень (рис. 4.6).

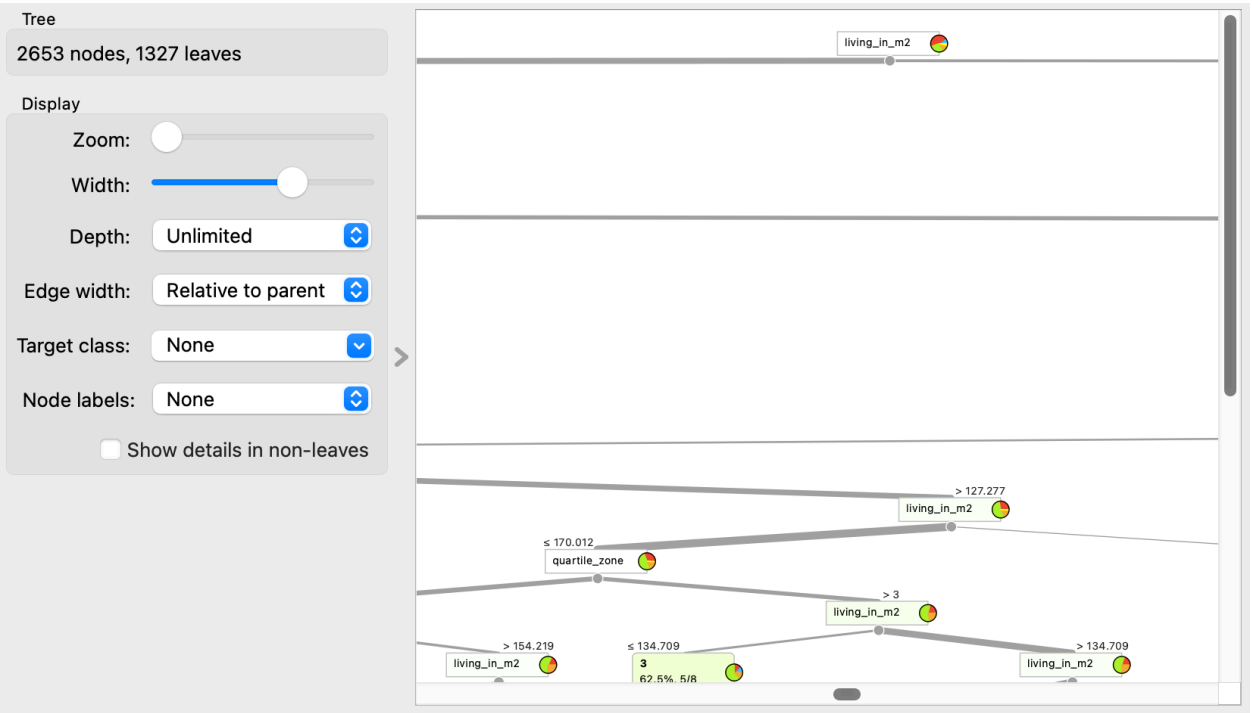


Рисунок 4.6 – Скріншот результатів з віджету Tree Viewer

Тепер візьмемо один об’єкт з вибірки і, згідно з структурою дерева рішень, продивимось просування даного об’єкта від кореневого вузла до листа, в якому об’єкт буде класифікований, дотримуючись умов переходу, що визначають по якому із ребер йти. Візьмемо об’єкт з рядка 24 нашої вибірки даних (табл. 4.1)

Таблиця 4.1 – дані вибраного об’єкту

price	bedrooms	grade	living_in_m2	real_bathrooms	quartile_zone
155000.0	1	1	93.83203	1	2

Для знаходження листа, який відповідає характеристикам нашого об’єкта, було зроблено 13 кроків. Знайдений лист було відносено до класу 2, оскільки частка об’єктів класу 2 в ньому становить 60% (рис. 4.7).

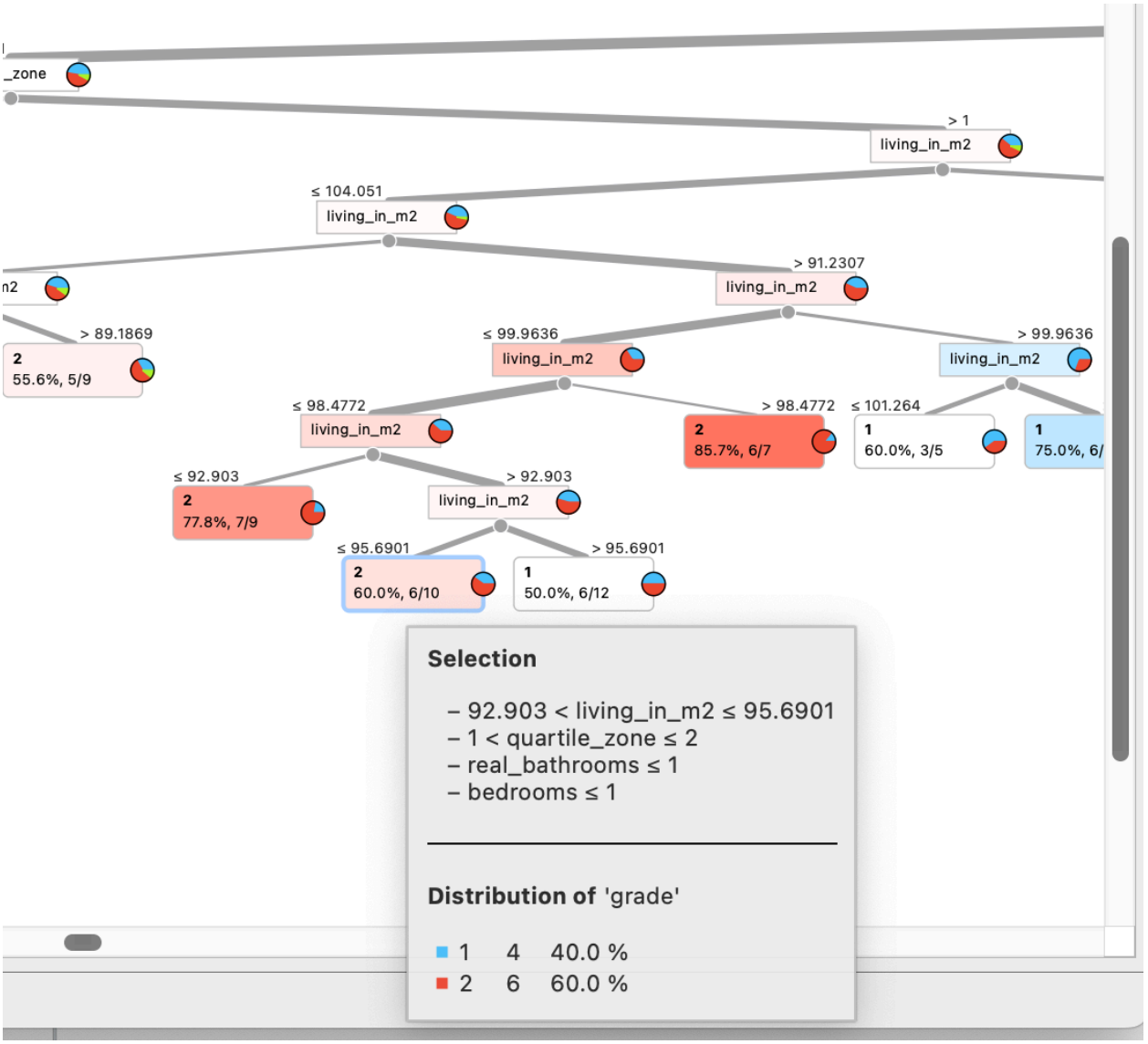


Рисунок 4.7 – Скріншот фрагменту дерева з віджету Tree Viewer (потрібний лист підсвічено синім та виділено його характеристики)

Виведемо результати роботи всіх класифікаторів за допомогою віджету Test and Score (рис. 4.8).

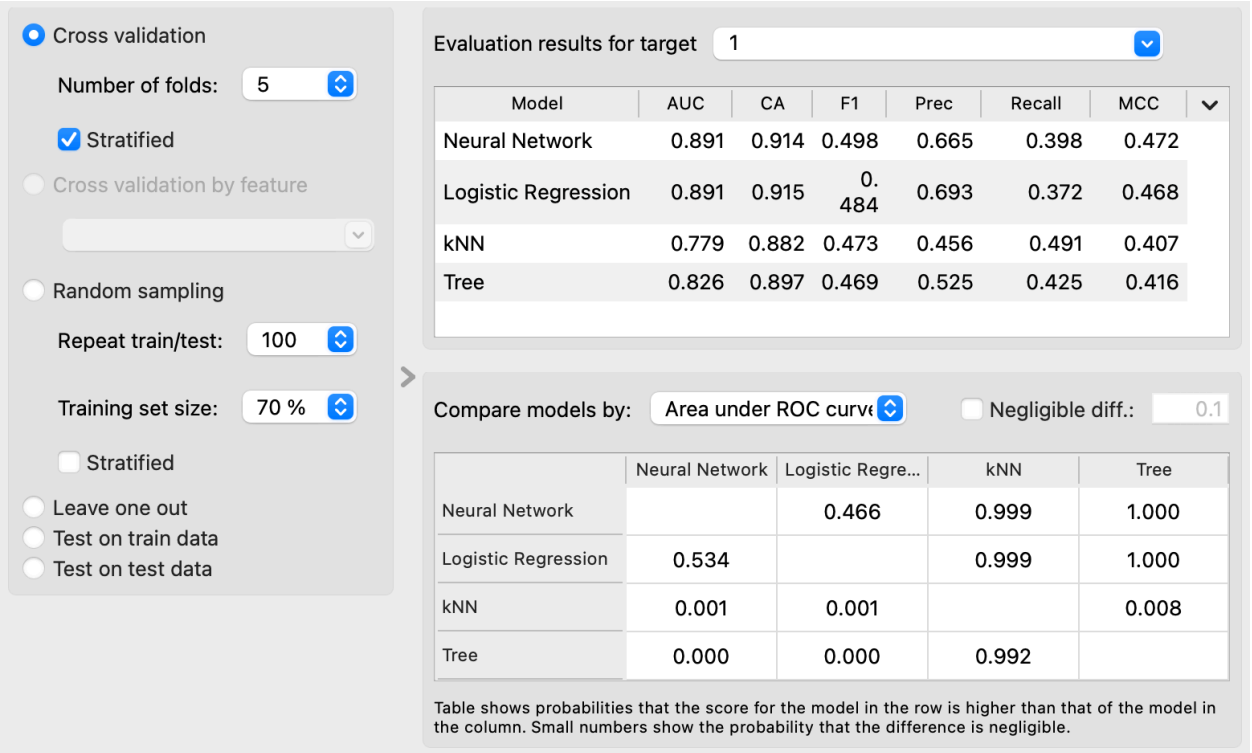


Рисунок 4.8 – Скріншот віджету Test and Score

Згідно з результатами модуля Test & Score, найкращу якість класифікації показала нейронна мережа, яка має найвищу метрику F_1 – 0.498, що свідчить про збалансовану точність і повноту. Інші моделі, зокрема дерево рішень та логістична регресія, показали нижчі показники через більшу кількість хибних спрацьовувань або пропусків позитивних випадків.

Підрахуємо помилки класифікації 1-го та 2-го роду та їх відносні частки істинно позитивних випадків та істинно негативних випадків, для кращого (нейронна мережа) та для гіршого (дерево рішень) класифікаторів, згідно з метрикою F_1.

Проводимо розрахунки для нейронної мережі:

Precision = 0.665 (істинно негативні) –> тобто 33.5% передбачених позитивів хибні –> це помилка 1-го роду

Recall = 0.398 (істинно позитивні) –> тобто 60.2% реальних позитивів пропущено –> це помилка 2-го роду

Проводимо розрахунки для дерева рішень:

Precision = 0.525 (істинно негативні) –> тобто 47.5% передбачених позитивів хибні –> це помилка 1-го роду

Recall = 0.425 (істинно позитивні) –> тобто 57.5% реальних позитивів пропущено –> це помилка 2-го роду

В результаті видно, що Neural Network краще розпізнає позитивні приклади точніші, але гірше покриває їх усі. Tree навпаки знаходить трохи більше позитивів (вищий Recall), але часто помиляється у позитивних передбаченнях.

Перейдемо до віджета ROC Analysis. Виведемо графіки ROC-кривих по одному графіку на клас (рис. 4.9-4.13).

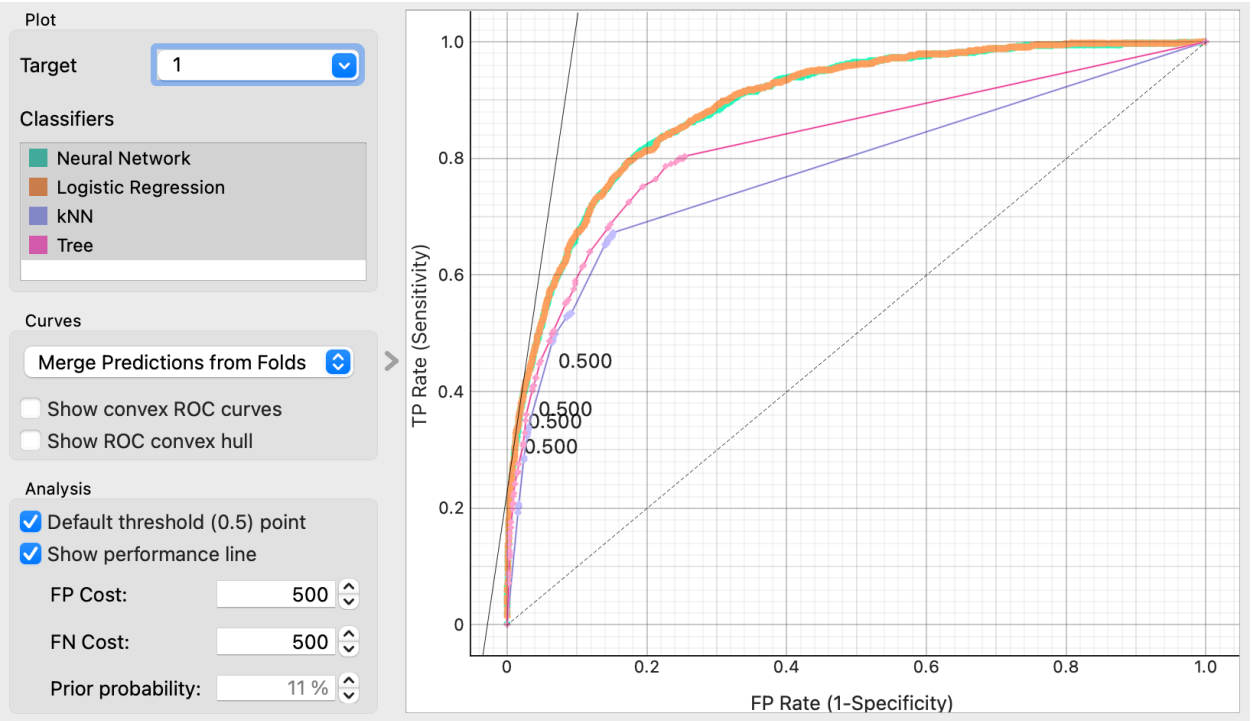


Рисунок 4.9 – Скріншот графіку ROC-кривих для класу 1

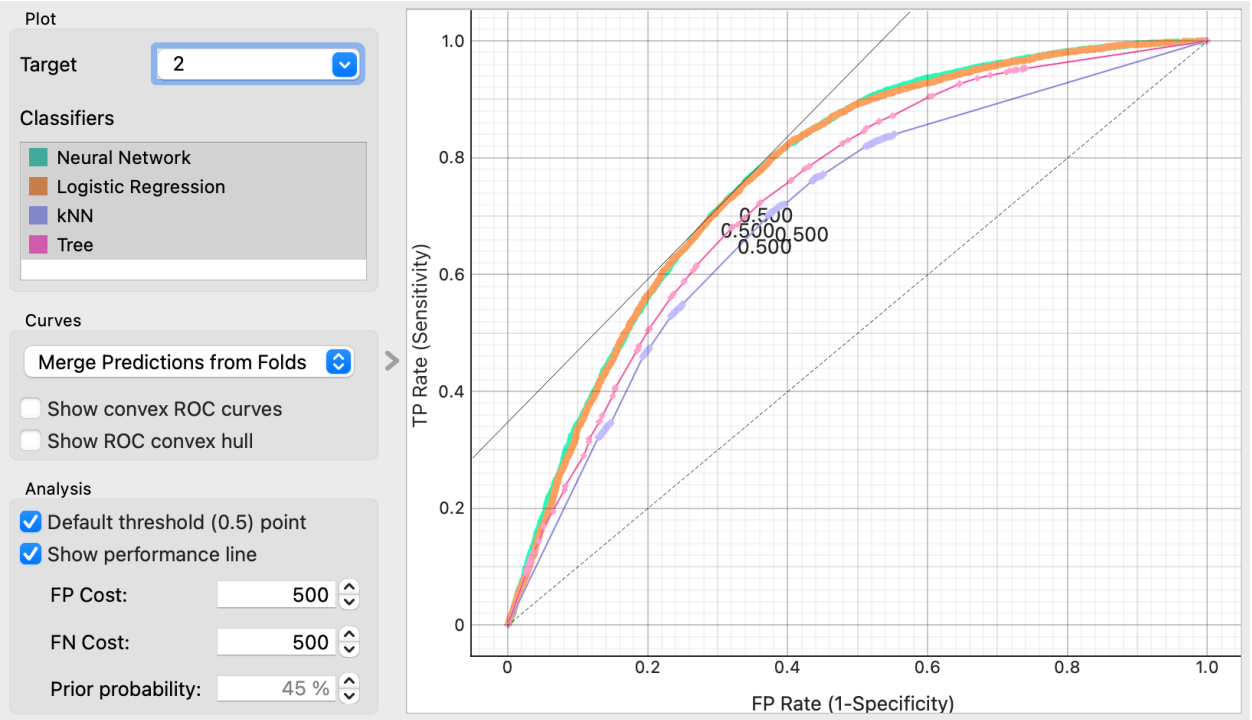


Рисунок 4.10 – Скріншот графіку ROC-кривих для класу 2

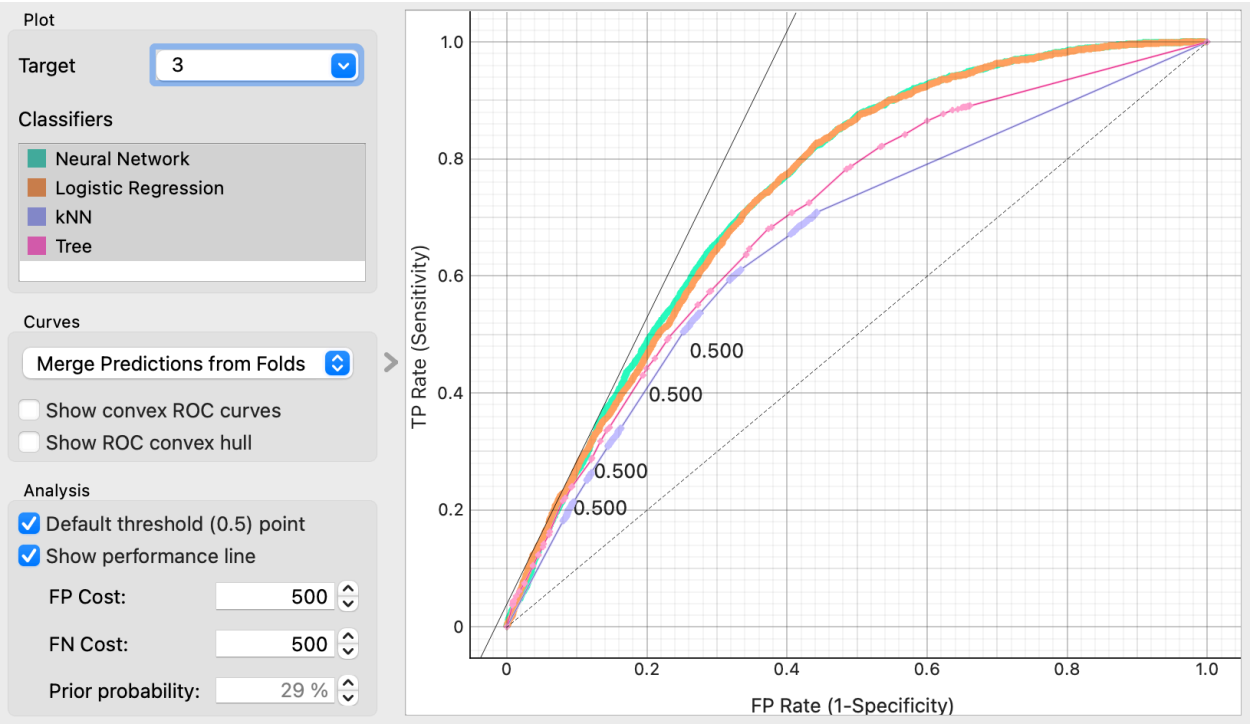


Рисунок 4.11 – Скріншот графіку ROC-кривих для класу 3

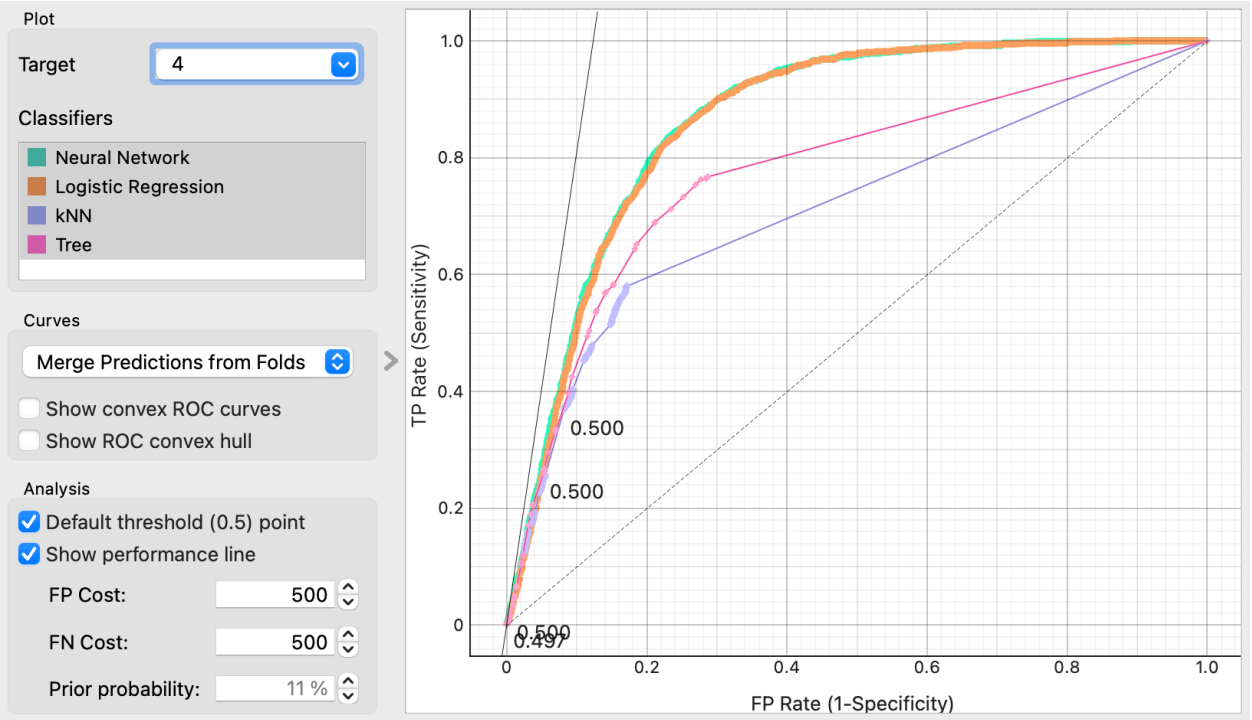


Рисунок 4.12 – Скріншот графіку ROC-кривих для класу 4

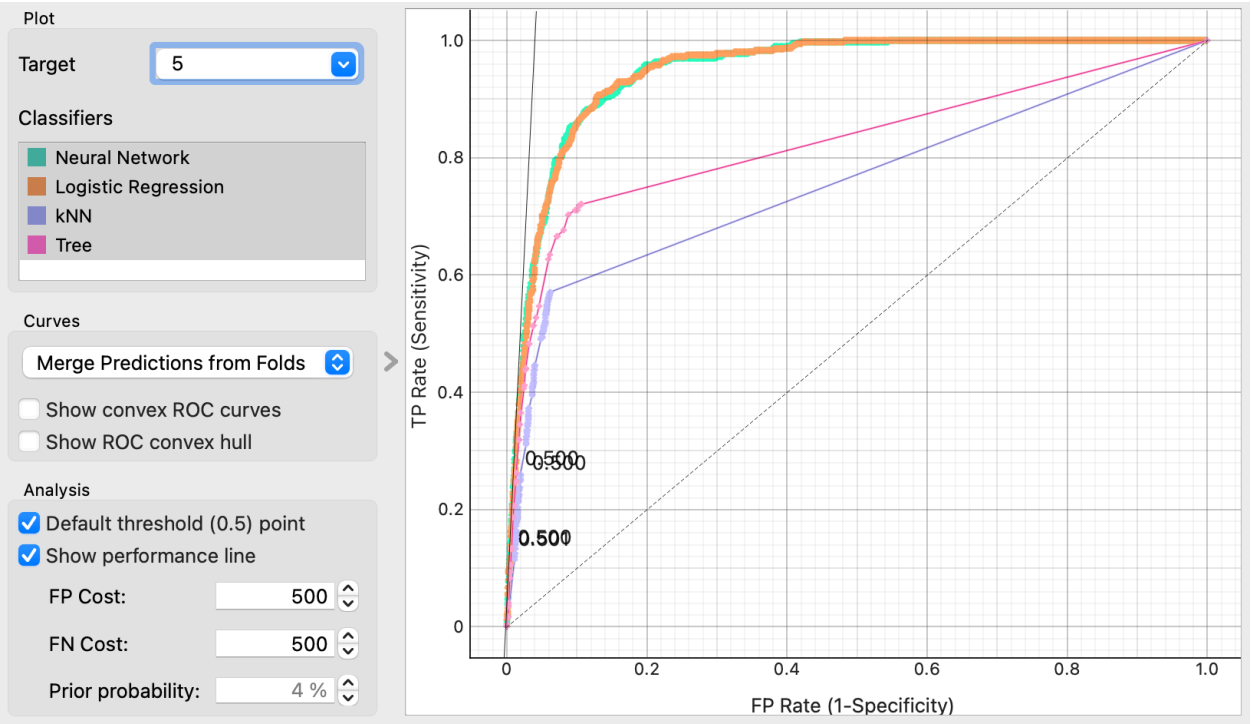


Рисунок 4.13 – Скріншот графіку ROC-кривих для класу 5

Графіки ROC-кривих показують, що класи 2 і 3 мають погану якість розпізнавання – моделі часто плутають позитивні та негативні приклади, що видно з форми кривих. Класи 4 і 5 демонструють кращі результати, але теж із

невисокою роздільною здатністю між класами. Клас 1 має середній рівень якості, що свідчить про часткову придатність моделей для виявлення цього класу, хоча загалом класифікація залишається нерівномірною між категоріями.

Продивимось результати віджету Confusion Matrix для найкращої класифікаційної моделі (Neural Network) при при параметрі Show = Number of Instances (рис. 4.14) та при Show = Proportion of Actual (рис. 4.15).

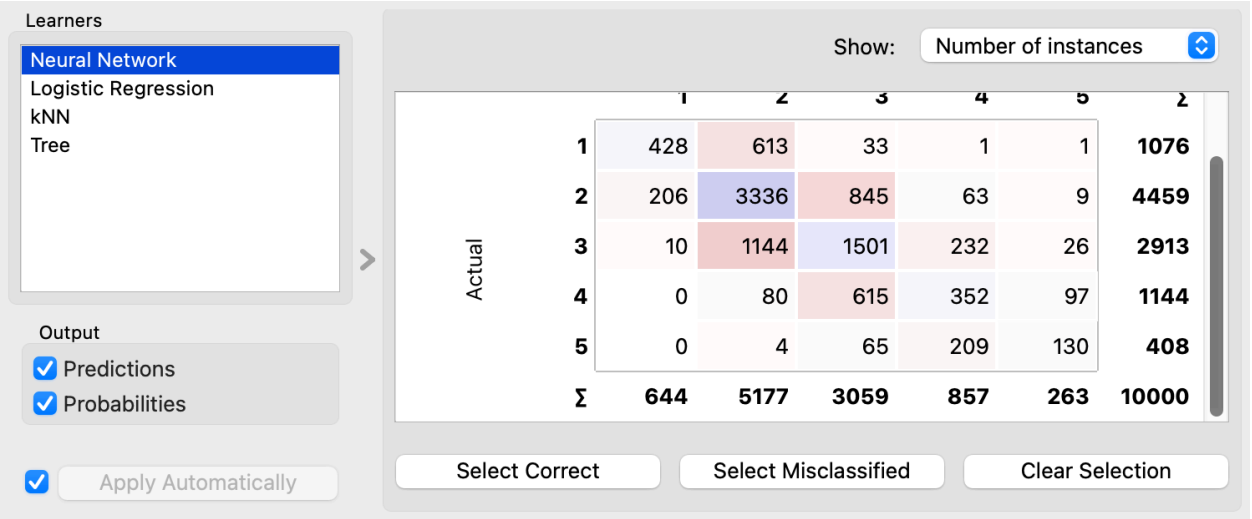


Рисунок 4.14 – Скріншот віджету Confusion Matrix для нейронної мережі при параметрі Show = Number of Instances

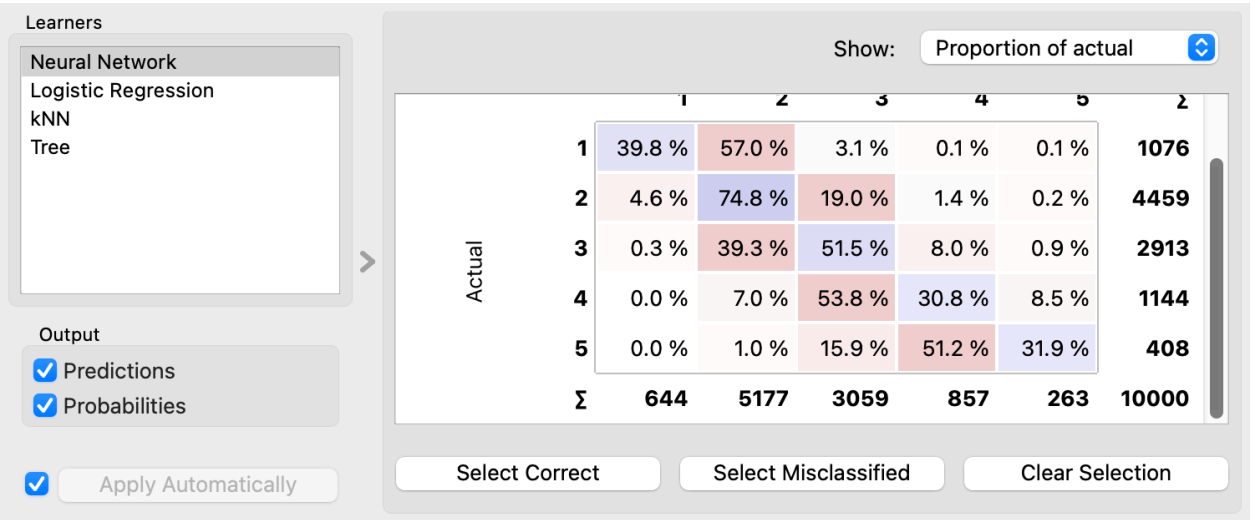


Рисунок 4.15 – Скріншот віджету Confusion Matrix для нейронної мережі при параметрі Show = Proportion of Actual

В результаті видно, що найкраща точність спостерігається для класу 2 (≈74.8%), тоді як класи 1 і 3 часто плутаються з класом 2, а класи 4 і 5 мають суттєве перекриття (30–50%). Отже, модель добре розпізнає другий клас, але потребує покращення роздільності між сусідніми або подібними класами.

Для інших моделей класифікації (Logistic Regression, kNN та Tree) збережемо лише скріншоти при параметрі Show = Proportion of Actual (рис. 4.16-4.18).

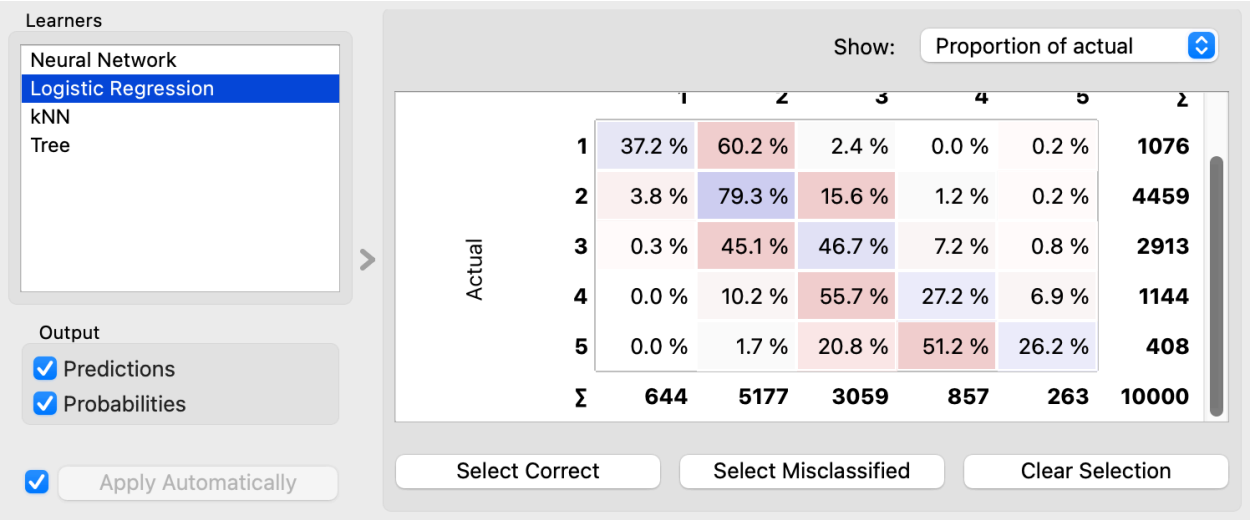


Рисунок 4.16 – Скріншот віджету Confusion Matrix для логістичної регресії при параметрі Show = Proportion of Actual

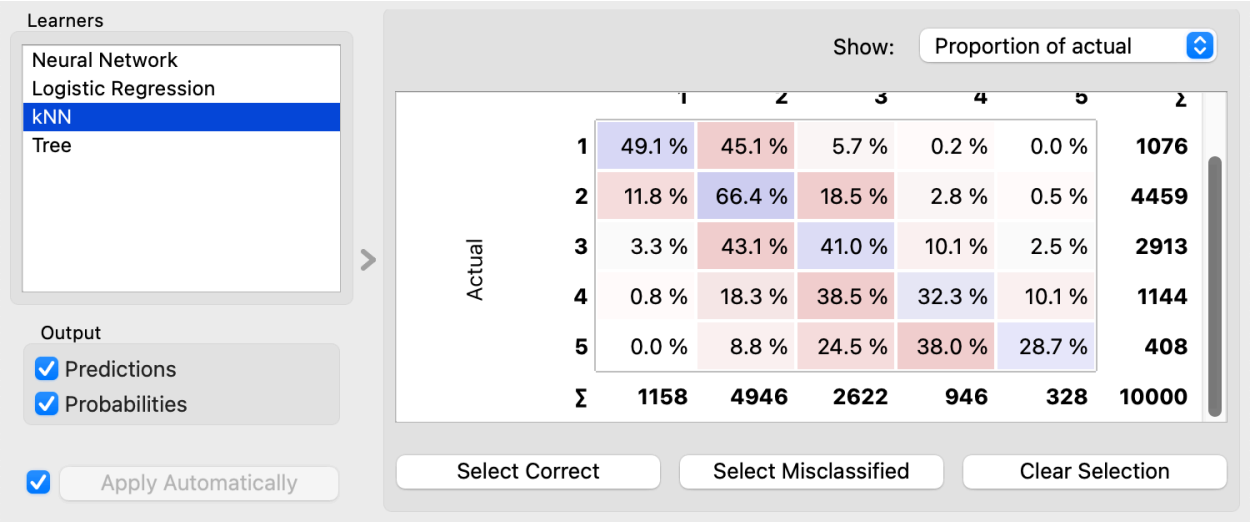


Рисунок 4.17 – Скріншот віджету Confusion Matrix для методу k найближчих сусідів при параметрі Show = Proportion of Actual

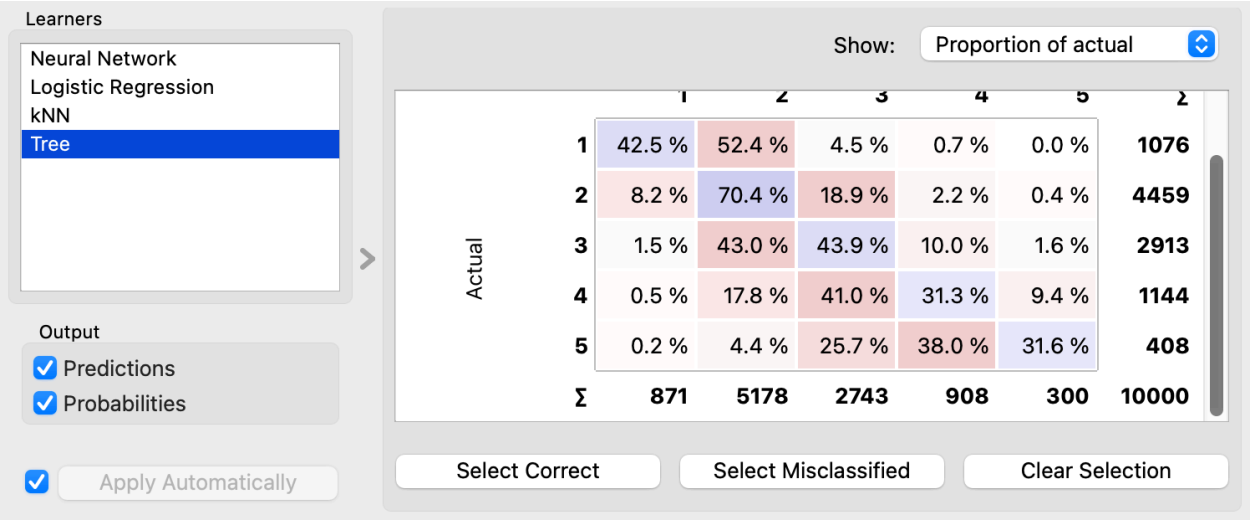


Рисунок 4.18 – Скріншот віджету Confusion Matrix для дерева рішень при параметрі Show = Proportion of Actual

Видно, що логістична регресія демонструє дещо вищу точність для класу 2 (79.3 % проти 74.8 % у нейромережі), проте загалом моделі мають схожу тенденцію плутати сусідні класи.

Висновки. У ході роботи було змодельовано задачу класифікації за допомогою середовища Orange, використовуючи різні алгоритми – нейронну мережу, логістичну регресію, kNN та дерево рішень. Проведено порівняння різних класифікаторів із використанням крос-валідації та аналізу ROC-кривих. За результатами експериментів встановлено, що найкращу якість класифікації продемонструвала нейронна мережа (за метрикою F1), хоча в окремих випадках логістична регресія не поступається їй у точності. Аналіз ROC-кривих підтвердив, що ці два алгоритми загалом майж ідентичні у якості класифікації, у той час як kNN та дерево рішень продемонстрували значно гірші результати. Загалом, всі моделі мали не ідеальні результати, що, скоріш за все, було пов’язано зі специфікою даних з виборки.