

Национальный исследовательский университет ИТМО
Факультет систем управления и робототехники

Прикладной искусственный интеллект
Лабораторная работа №1

Выполнил:

Никандров Сергей Андреевич

R32352

Преподаватель:

Евстафьев Олег Александрович

Санкт-Петербург

2022 год

Цель лабораторной: необходимо по данным с мобильных сенсоров при помощи прикладных алгоритмов машинного обучения предсказать активность человека по шести классам движений:

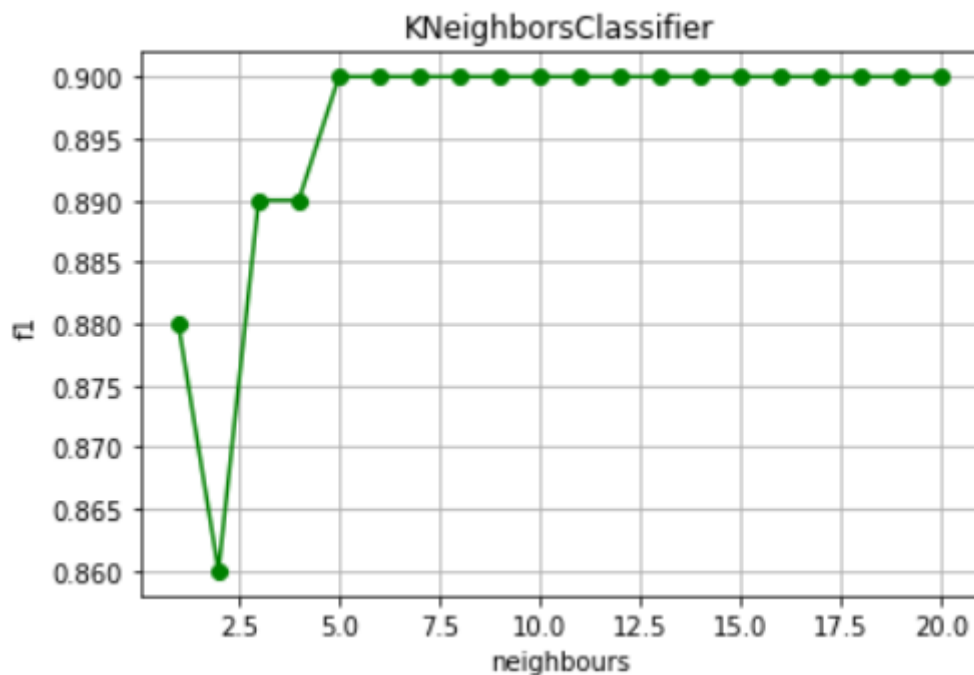
- Двигается по прямой
- Двигается вверх (например, движение по лестнице вверх)
- Двигается вниз (например, движение по лестнице вниз)
- Сидит
- Стоит
- Лежит

Иначе говоря, у нас есть данные, которые были сняты с помощью датчиков и нам нужно на основе их определять активность человека.

Пояснение выбранных моделей:

1. Метод К-ближайших соседей. Алгоритм для автоматической классификации объектов или регрессии (зависимость, устанавливающая соответствие между случайными переменными). Мы используем для классификации. В случае метода для классификации объект присваивается тому классу, который является наиболее распространённым среди k соседей данного элемента, классы которых уже известны. Для того, чтобы определить при каком гиперпараметре достигается лучшая точность я запустил программу на разных количествах соседей и построил график зависимости точности от количества

соседей. И из него можно понять, что максимальная точность, которую можно получить с помощью данного метода равна 0.9.

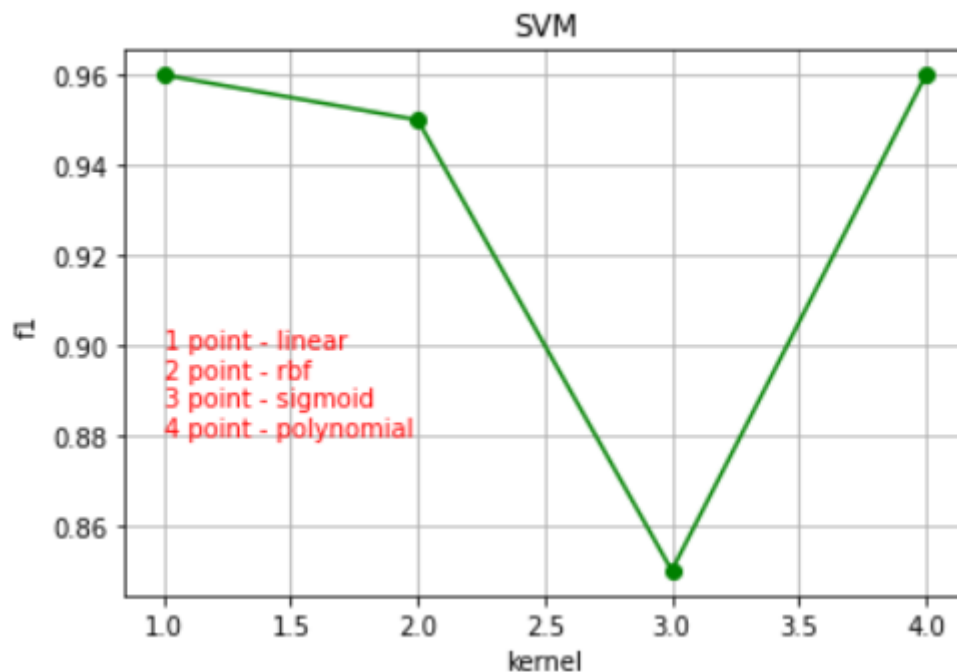


2. Метод опорных векторов. Основной задачей алгоритма является найти наиболее правильную линию, или гиперплоскость разделяющую данные на классы. Алгоритм SVM устроен таким образом, что он ищет точки на графике, которые расположены непосредственно к линии разделения ближе всего. Эти точки называются опорными векторами. Затем, алгоритм вычисляет расстояние между опорными векторами и разделяющей плоскостью. Это расстояние, которое называется зазором. Основная цель алгоритма — максимизировать расстояние зазора. Лучшей гиперплоскостью считается такая гиперплоскость, для которой этот зазор является максимально большим. При создании нашего метода мы можем выбрать одну из нескольких функций:

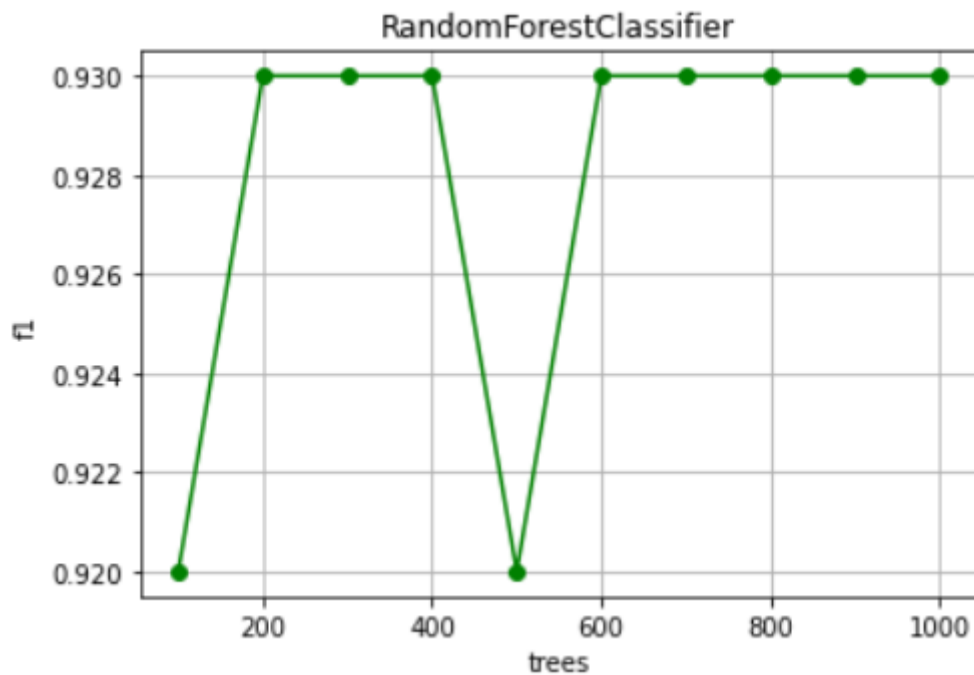
- linear: $\langle x, x' \rangle$.
- polynomial: $(\gamma \langle x, x' \rangle + r)^d$, where d is specified by parameter `degree`, r by `coef0`.
- rbf: $\exp(-\gamma \|x - x'\|^2)$, where γ is specified by parameter `gamma`, must be greater than 0.
- sigmoid $\tanh(\gamma \langle x, x' \rangle + r)$, where r is specified by `coef0`.

polynomial, rbf(radial basis function) и sigmoid особенно полезны, когда данные нельзя разделить линейно. Я также перебрал все функции и построил график для определения лучшей.

И из графика видно, что данный метод с линейной или полиномиальной функцией лучше, чем метод k-ближайших соседей и точность равна 0.96.



3. Метод случайного леса. универсальный алгоритм машинного обучения, суть которого состоит в использовании ансамбля решающих деревьев. Само по себе решающее дерево предоставляет крайне невысокое качество классификации, но из-за большого их количества результат значительно улучшается. Для определения лучшего количества деревьев я их перебрал и построил график зависимости. Из него понятно, что максимальную точность, которую мы можем получить – 9.3.



Параметры

Precision – метрика, отражающая отношение правильно выбранных объектов ко всем объектам, которые должны были быть выбраны.

Recall – метрика, отражающая отношение правильно выбранных объектов ко всем объектам, которые должны были быть выбраны.

Accuracy – отражает точность модели, то есть отношение правильных предсказаний ко всем.

F1-score – метрика, объединяющая recall и precision в одно значение, некое среднее своего рода. Среднее гармоническое между precision и recall.

Выводы лучших параметров для каждой модели:
Метод К-ближайших соседей:

	precision	recall	f1-score	support
Walking	0.85	0.98	0.91	496
Walking Upstairs	0.89	0.90	0.90	471
Walking Downstairs	0.95	0.79	0.86	420
Sitting	0.91	0.79	0.85	491
Standing	0.83	0.93	0.88	532
Laying	1.00	0.99	1.00	537
accuracy			0.90	2947
macro avg	0.91	0.90	0.90	2947
weighted avg	0.91	0.90	0.90	2947

Метод опорных векторов:

	precision	recall	f1-score	support
Walking	0.96	0.99	0.97	496
Walking Upstairs	0.98	0.96	0.97	471
Walking Downstairs	0.99	0.98	0.98	420
Sitting	0.96	0.89	0.92	491
Standing	0.91	0.97	0.94	532
Laying	1.00	1.00	1.00	537
accuracy			0.96	2947
macro avg	0.97	0.96	0.96	2947
weighted avg	0.96	0.96	0.96	2947

Метод случайного леса:

	precision	recall	f1-score	support
Walking	0.91	0.97	0.94	496
Walking Upstairs	0.89	0.92	0.91	471
Walking Downstairs	0.96	0.85	0.90	420
Sitting	0.91	0.90	0.90	491
Standing	0.91	0.92	0.91	532
Laying	1.00	1.00	1.00	537
accuracy			0.93	2947
macro avg	0.93	0.93	0.93	2947
weighted avg	0.93	0.93	0.93	2947

Ссылка на код:

<https://datalore.jetbrains.com/notebook/kJEwvIMafpjQ3KeI9h5Pig/oLvPEaBwyEcZ5pVfC6NaT0/>

Выводы

В ходе проделанной работы можно сделать вывод о том, что для нашей задачи лучше всего подходит метод опорных векторов с ядром в виде линейной или полиномиальной функции.