|  |  |  |  |
| --- | --- | --- | --- |
| Tên thuật toán | Ưu điểm | Nhược điểm |  |
| Naïve Bayes | + Dễ dàng cài đặt.  + Thời gian thi hành nhanh  + Đạt kết quả tốt trong phần lớn các trường hợp | + Giả thiết về tính độc lập điều kiện của các thuộc tính làm giảm độ chính xác |  |
| Binary Classifiers | +  Không cần có giả thiết dữ liệu hai class là linearly separable.  + Dễ áp dụng vào bài toán.  + Thời gian thi hành nhanh.  + Thích hợp dùng cho bài toán phân lớp | + Chỉ phù hợp với loại dữ liệu mà hai class là gần với linearly separable.  + Không làm việc được với kiểu dữ liệu phi tuyến.  + Yêu cầu các điểm dữ liệu được tạo ra một cách độc lập với nhau. |  |
| K-nearest neighbors | + Độ phức tạp tính toán của quá trình training là bằng 0.  + Việc dự đoán kết quả của dữ liệu mới rất đơn giản.  + Không cần giả sử gì về phân phối của các class | + KNN rất nhạy cảm với nhiễu khi K nhỏ.  + Việc tính khoảng cách tới từng điểm dữ liệu trong training set sẽ tốn rất nhiều thời gian, đặc biệt là với các cơ sở dữ liệu có số chiều lớn và có nhiều điểm dữ liệu.  + Việc lưu toàn bộ dữ liệu trong bộ nhớ cũng ảnh hưởng tới hiệu năng của KNN. |  |
| Softmax Regresion | + Có thể giải quyết Các bài toán classification có nhiều class.  + Sử dụng tốt trong các mạng Neural có nhiều lớp. Thường được ứng dụng ở lớp cuối cùng của mô hình.  + Không cần có giả thiết dữ liệu các class là linearly separable. | + Chỉ phù hợp với loại dữ liệu mà các class là gần với linearly separable. |  |
| Multi-layer Perceptron and Back-propagation | + Không giới hạn dữ liệu thuộc linear hay không.  + Boundary phụ thuộc vào dữ liệu train.  + Đạt kết quả tốt trong phần lớn các trường hợp.  + Có thể mở rông, kết hợp thêm các thuật toán khác để đạt được kết quả mong muốn. | + Gặp khó khăn khi dữ liệu rời rác.  + Tốn nhiều thời gian để tìm ra được số lượng units, hidden layers thích hợp.  + Khi số lượng hidden layers lớn lên, số lượng hệ số cần tối ưu cũng lớn lên và mô hình sẽ trở nên phức tạp. Dẫn đến tốc độ chậm đi. Dẫn đến hiện tượng Overfitting. |  |
| Support Vector Machine | + Không cần xác định mô hình của đối tượng như neuron, fuzy logicm,…  + Giải quyết tốt bài toán có dữ liệu có số chiều lớn.  + Giải quyết vấn đề overfitting rất tốt.  + Phân lớp nhanh, hiệu suất tổng hợp tốt, hiệu suất tính toán cao, **tiết kiệm bộ nhớ, tính linh hoạt** | + Độ phức tạp cao.  + Hạn chế khi xử lý dữ liệu kiểu số.  + Dể SVM hiệu quả thì dữ liệu cần phải được trích-chọn các thuộc tính phù hợp, SVM không thể lựa chọn thuộc tính được nên phải tự lựa chọn thuộc tính hoặc dùng các thuật toán khác.  + Trong trường hợp số lượng thuộc tính (**p**) của tập dữ liệu lớn hơn rất nhiều so với số lượng dữ liệu (**n**) thì SVM cho kết quả khá tồi.  + **Chưa thể hiện rõ tính xác suất** |  |
| Decision trees | + Reasonable training time  + Fast application  + Easy to interpret  + Easy to implement  + Can handle large number of features.  + Does not require assumption of unidiménionality or local indêpndence | + Cannot handle complicated relatiónhip between features  + simple decision boundaries  + problem with lots of missing data |  |
|  |  |  |  |