Сравнение алгоритмов классификации на предмет устойчивости к зашумлению входных данных

Владимир Назаренко
Научный руководитель:
д.ф.-м.н., проф. Б.А. Новиков

Санкт-Петербургский Государственный Университет
Математико-механический факультет

Санкт-Петербург
2014

Обзор предметной области

- Машинное обучение находит своё применение в большом количестве задач
- Обучение с учителем
- Классификация
- Данные часто ошибочны
- Некорректные данные влекут увеличение количества ошибок классификации
- Средство анализа данных Weka

Постановка задачи

- Выполнить сравнение качества работы некоторых алгоритмов классификации при наличии шумов в тестовом множестве, где мерой качества является количество верно классифицированных элементов в тестовом множестве
- Провести анализ изменений качества работы алгоритмов при увеличении интенсивности шума и провести анализ причин изменений качества работы

Работы в данной области

- Ассиметричное зашумление аттрибутов(Mannino et al. 2009)
- Деградация алгоритма "наивный байес" на зашумленных данных(Glick et al. 2004)

Описание классификаторов (1)

Классификатор на основе дерева выбора

- Дерево выбора
- Применение дерева выбора в классификации
- Проблема выбора разделения
- Random Tree

- Теорема Байеса $P(y = C|x) = \frac{P(C)P(x|y=C)}{P(x)}$
- Следствия независимости свойств объекта

$$P(y = C|x) = \frac{P(C)\prod_{i=1}^{n} P(x_i|y = C)}{\prod_{i=1}^{n} P(x_i)}$$

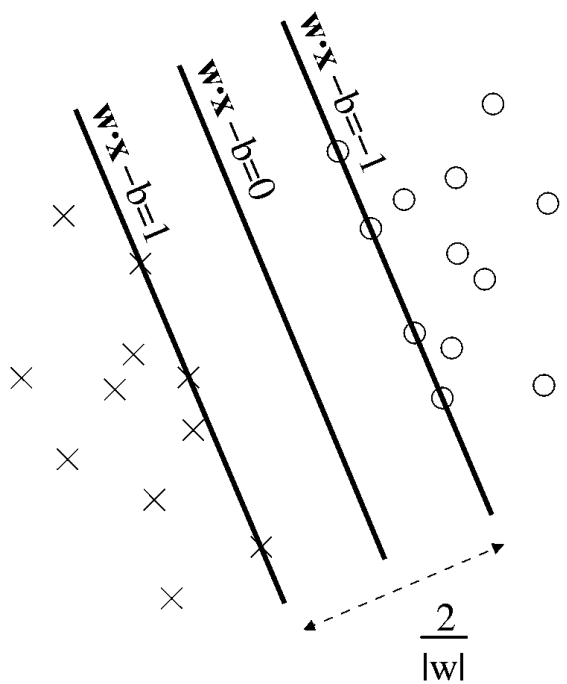
$$\max_{C \in \{C_1,..,C_n\}} P(y = C|x)$$

- Naive Bayes

Описание классификаторов (3)

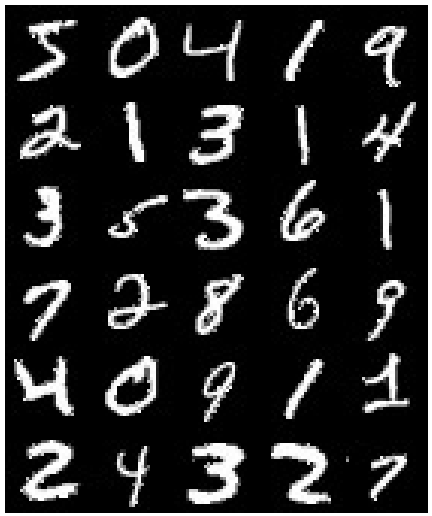
Классификатор на основе метода опорных векторов

- Вектор свойств объекта представляется точкой
- Нужно найти лучшую разделяющую плоскость
- Проблема максимального запаса
- SMO



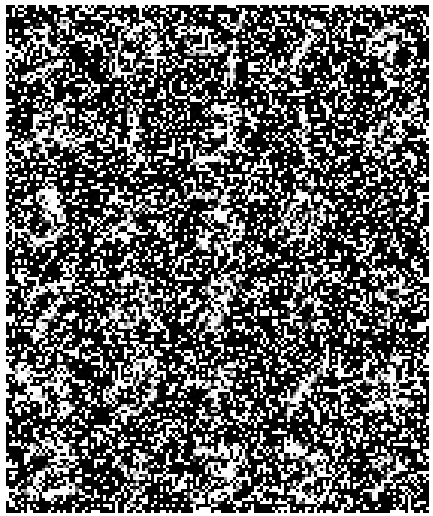
Описание эксперимента

- В качестве набора данных был выбран MNIST Database
- На вектор свойств наносился шум случайным образом



Изображения без шума

30% пикселей испорчены

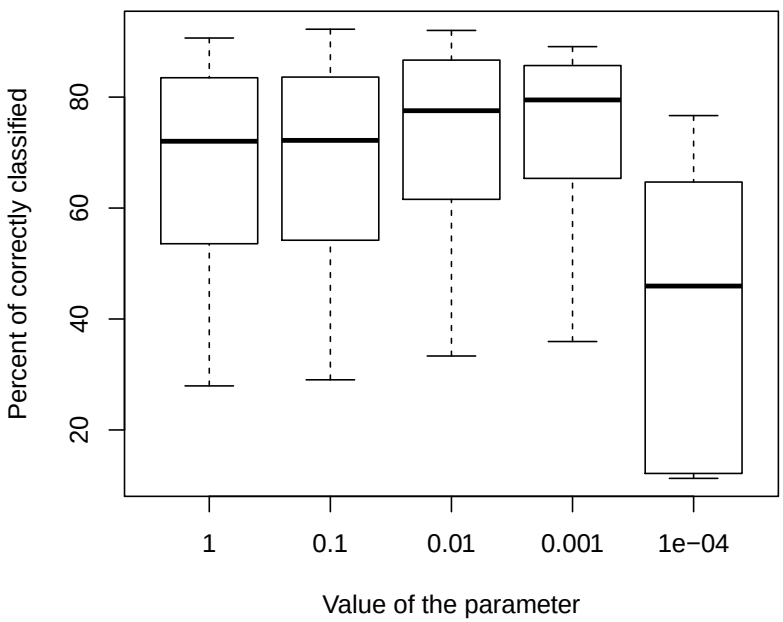


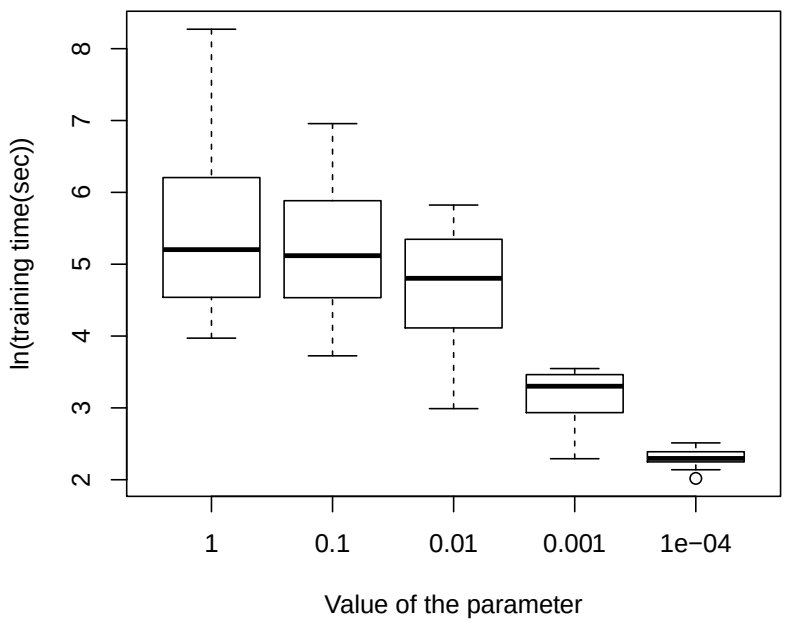
60% пикселей испорчены

- Выбранные алгоритмы тестировались в двух ситуациях
  - Зашумлено только тестовое множество
  - И тестовое, и тренировочное множество зашумлены

Технические детали

- Для нанесения шумов и конвертации набора данных использовался скрипт на Python
- Все параметры алгоритмов были оставлены стандартными, кроме параметра complexity для SMO
- Для SMO параметр complexity был подобран





Результаты эксперимента(1)

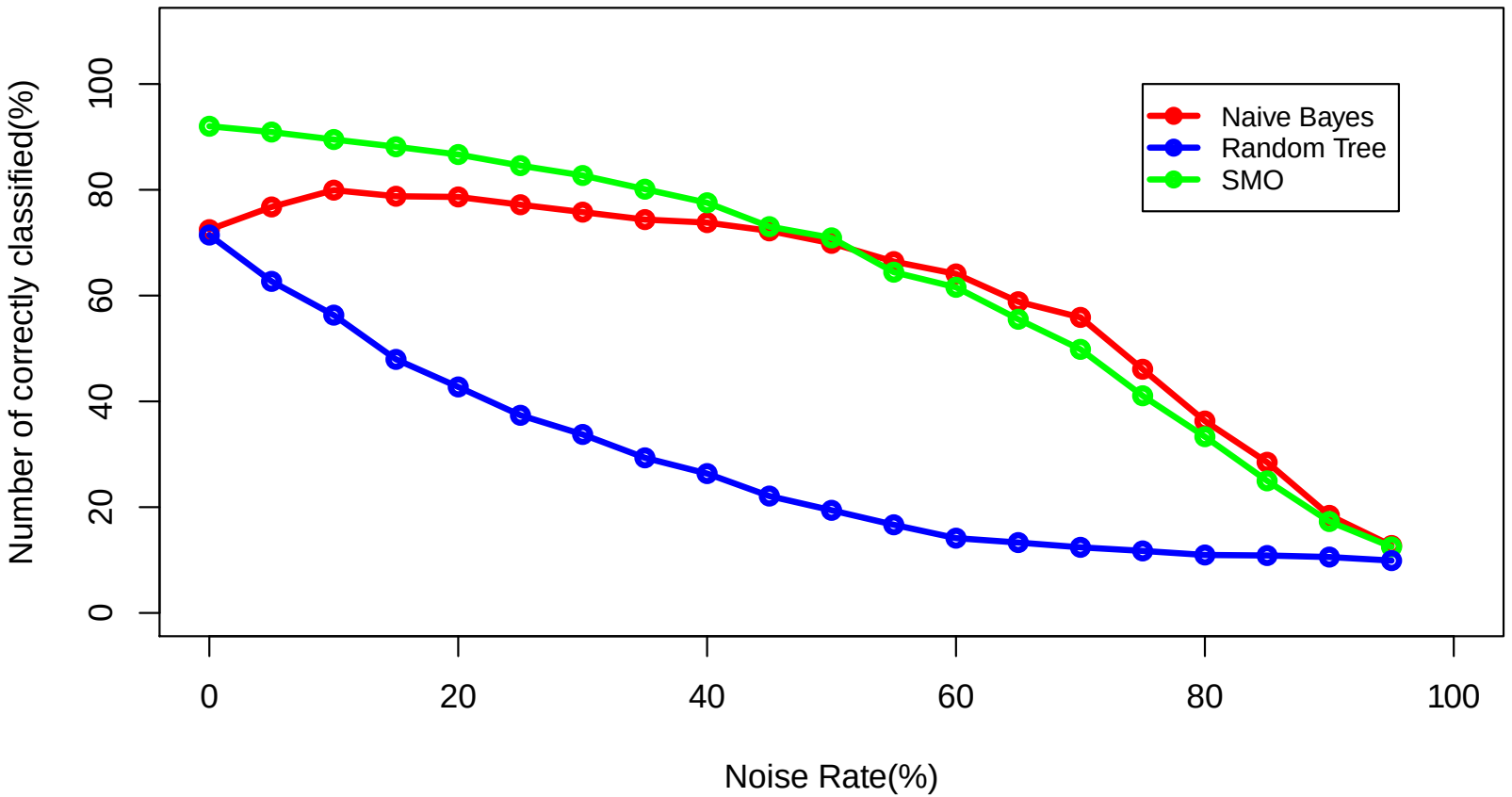


Figure : Эффективность алгоритмов при построении модели на зашумленных данных

Результаты эксперимента(2)

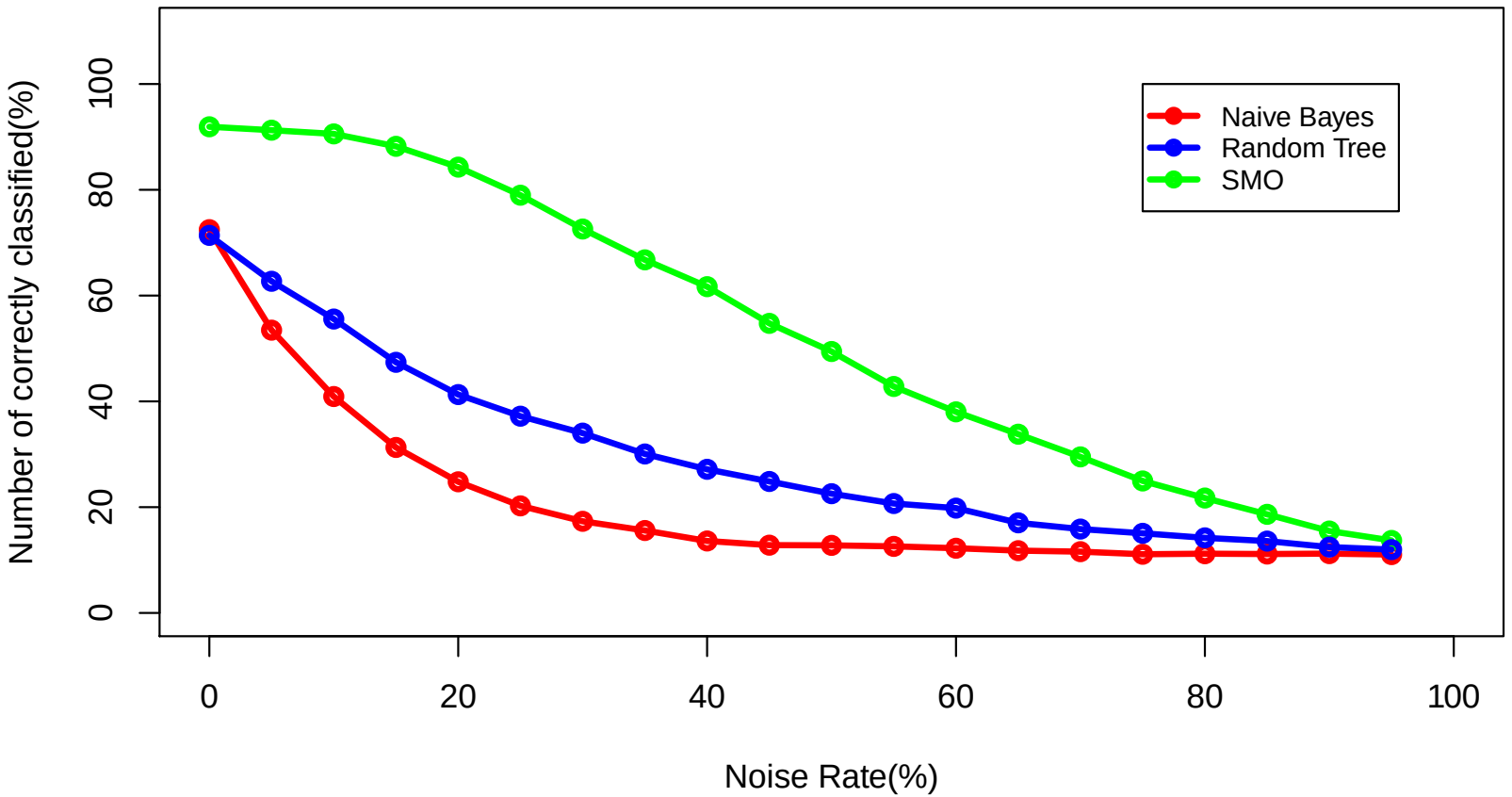


Figure : Эффективность алгоритмов при построении модели на незашумленных данных

Выводы

- В работе было произведено сравнение алгоритмов Naive Bayes, SMO и Random Tree на предмет устойчивости к зашумлению входных данных
- Алгоритм “Random Tree” оказался неустойчив к шуму
- Алгоритм “Naive Bayes” показал удовлетворительные результаты при обучении на зашумленном множестве и неудовлетворительные при обучении на незашумленном
- Классификатор “SMO” показал лучшие результаты из представленных алгоритмов

Результаты

- Выполнено сравнение трёх популярных алгоритмов классификации на предмет устойчивости к шумам в тестовом множестве
- Проведён анализ причин устойчивости/неустойчивости алгоритмов к шумам
- Построено тестовое окружение, позволяющее легко воспроизвести эксперимент