Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Новосибирский государственный технический университет»

Кафедра теоретической и прикладной информатики

**Отчет ПО ПРАКТИКЕ**

Производственная практика: практика по получению профессиональных умений и опыта профессиональной деятельности

Направление подготовки: Математическое обеспечение и администрирование информационных систем.

|  |  |
| --- | --- |
| Выполнил:  Студент Паничев Сергей Евгеньевич  (Ф.И.О.)  Группа ПМИ-31  Факультет прикладной математики и информатики  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  подпись  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2017 г. | Проверил:  Руководитель от НГТУ Попов Александр Александрович  (Ф.И.О.)  Балл: \_\_\_\_\_\_\_\_\_\_, ECTS\_\_\_\_\_\_\_\_\_\_\_\_\_,  Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  «отлично», «хорошо», «удовлетворительно», «неуд.»  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  подпись  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2017 г. |

Новосибирск 2017

Оглавление

[Введение 3](#_Toc482348879)

[Одноядерный LS SVM 4](#_Toc482348880)

[Многоядерный LS SVM 5](#_Toc482348881)

[Алгоритм 6](#_Toc482348882)

[Список использованных источников 7](#_Toc482348883)

# Введение

В течение последних нескольких лет ядерные методы, в частности метод опорных векторов (SVM), как выяснилось, являются эффективными инструментами для изучения и решения проблем регрессионного анализа. [1]

Эффективность использования ядер заключается в том, что выбранное исследователем ядро позволяет методу действовать в неявном пространстве признаков высокой размерности даже без явного вычисления координат данных в исходном пространстве признаков, просто вычисляя скалярное произведение отображений всех пар данных в этом пространстве. Ядро, таким образом, является функцией сходства над парами точек в исходном представлении. Эта операция часто является вычислительно менее затратной, чем явный расчёт координат. Данных подход называют «kernel trick». [2]

Первоначальный метод опорных векторов для решения задач классификации и оценивания нелинейных функций был изобретён Владимиром Вапником и Алексеем Червоненкисом в 1963 году [3], текущее стандартное воплощение было опубликовано в 1995-м году [4]. За многие годы данный метод был исследован различными учёными и является важной методологией в области нелинейного моделирования. В то время, как классические подходы нейронных сетей страдают в частности от такой проблемы, как существование множества локальных минимумов, в SVM решается задача квадратичного программирования, которая имеет единственное решение. В 1996 году SVM был обобщен В. Вапником, Н. Дракером, К. Берджесом, Л. Кауфманом и А. Смола для оценивания действительных функций [5].

В SVM минимизирована верхняя граница ошибок обобщения (иными словами, вероятность неправильной классификации), что решает проблему переобучения. В процессе обучения SVM решается задача оптимизации квадратичной функции, зависящей от нескольких переменных с накладываемыми на эти переменные линейными ограничениями для получения глобального оптимального решения. Тем не менее, есть некоторые вопросы, которые обычный SVM не позволяет решить. Одним из таких вопросов является выбор ядер [6]. Но ещё более сложным вопросом является вычислительная сложность [7], которая значительным образом влияет на скорость расчётов. В данном методе большое количество времени занимает решение проблемы квадратичной оптимизации. Особенно заметна данная проблема при обработке больших данных.

Множество различных модификаций SVM было предложено, чтобы решить данную проблему [8-10]. Однако в 1999 году, Дж. Сайкенсом было предложено расширение SVM с использованием квадратичной функции потерь [11], которое получило название LS SVM, который и был исследован в ряде научных работ [12-14].

Решение по методу LS SVM находится как решение системы линейных уравнений, полученных из условий оптимальности Куна-Таккера, в то время, как решение по методу SVM предусматривает решение задачи квадратичного программирования. Данный алгоритм не только лёгкий в плане реализации, при этом он является гораздо более простым с точки зрения алгоритмической сложности и, как следствие, позволяет значительно ускорить вычисления.

Упрощение вычисления позволяет вернуться к ключевому вопросу, решение которого позволит в значительной степени повысить точность оценки - к выбору ядер и соответствующих им параметрам [15]. К сожалению, нету никаких чётких правил по подбору ядер для той или иной ситуации. Данный вопрос можно решить только прибегнув к исследованиям.

Метод опорных векторов имеет широкое распространение в самых разных областях науки. С помощью метода опорных векторов можно давать оценку кредитного риска для предприятий [16], прогнозировать тенденции отказа или работоспособности систем, прогнозировать риски при проведении операций, производить трекинг 3D объектов с помощью SVR.

Основной целью данной работы является изучение и программная реализация многоядерного метода LS SVM, позволяющего производить оценку функций, гибкую настройку ядер для получения максимальной точности оценки, а также разработка метода оценка качества алгоритма, основанного на методе наименьших квадратов.

# Одноядерный LS SVM

Рассмотрим набор тренировочных данных, состоящий из  независимых наблюдений c входными данными (аргументами) , и целевыми значениями функции.

LS SVM должен построить модель следующего вида:



где  функция отображения на высокоразмерное пространство признаков, которое потенциально может являться бесконечной размерности,  - весовой вектор.

Согласно структурному принципу минимизации риска [17, 18], проблема поиска  и может быть представлена как решение следующей ограниченной оптимизационной задачи:



таким образом, что 

где , - это весовая функция, - параметр регуляризации и значение ошибки.

Чтобы решить вышеупомянутую ограниченную задачу оптимизации, определим функционал Лагранжа следующим образом:



где - вектор множителей Лагранжа.

Поскольку оптимальное решение (0.2) удовлетворяет условиям Крауша-Куна-Такера (KKT), то оптимальные условия выражены следующим образом:



После исключения и получаем следующую СЛАУ:



где , , - n-мерный единичный вектор,  - единичная матрица, .

Из данной СЛАУ получаем следующие уравнения:



Отсюда получаем формулы для вычисления множителей Лагранжа и :



Результирующая LS SVM модель для оценки функции будет иметь следующий вид:



где  и  являются решениями СЛАУ (0.5).

# Проверка работоспособности

# Многоядерный LS SVM

Многоядерность алгоритма LS SVM заключается в использовании в качестве ядерной функции линейной комбинации всех ядер:

,

где m – количество используемых ядер, , , при следующих условиях:



Итоговая модель LS SVM для оценки функции (0.8) преобразуется следующим образом:



где .

Таким образом, из формулы (0.5) пример в многоядерной реализации следующий вид:



Оптимальные коэффициенты получаются в результате решения следующей задачи квадратичной оптимизации:



где , , .

# Алгоритм

Действия, необходимые для реализации MK LS SVM:

1. *Инициализация*: задание начального значения параметра регуляризации , Коэффициенты  задаются как: ;
2. *Получение множителей Лагранжа :* множители Лагранжа вычисляются в результате решения СЛАУ, описанного в формуле (1.7), где значения коэффициентов при ядрах равны ;
3. *Получение коэффициентов ядер :* коэффициенты при ядрах  вычисляются посредством решения задачи квадратичного программирования, описанного в формуле (1.11), основанного на множителях Лагранжа, полученных на последней итерации алгоритма;
4. *Вычисление ошибок:* если результаты не сходятся, то возврат к шагу 2 и реализация двухэтапной оптимизации снова. Если результаты неудовлетворительны, то возврат к нашу 1 к настройке параметра регуляризации.

# **Список использованных источников**

1. **Scholkopf, B., & Smola, A. (2001). Learning with kernels. MIT Press.**
2. <https://en.wikipedia.org/wiki/Support_vector_machine>**.**
3. **Vapnik V. Pattern recognition using generalized portrait method / V. Vapnik, A. Lerner, Automation and Remote Control, 1963. – vol. 24, – no. 6, – pp. 774 – 780.**
4. **Corinna Cortes, Vladimir Vapnik, Support-vector networks, 1995, Volume 20, Issue 3, pp 273–297**
5. **Support vector regression machines / H. Drucker, C. J. C. Burges, L. Kaufman, A. Smola, Eds. // In M. Mozer, M. Jordan, and T. Petsche, editors, Advances in Neural Information Processing Systems, – Cambridge: MA, MIT Press, 1997. – vol. 9, – pp. 155 – 161**
6. **C. S. Onq, A. J. Smola and R. J. Williamson, Learning the kernel with hyperkernels, Journal of Machine Learning Research 6 (2005) 1043–1071.**
7. **B. Scholkopf and A. Smolla, Learning with Kernels (MIT Press, Cambridge, USA, 2002).**
8. **R. Debnath, M. Muramatsu and H. Takahashi, An efficient support vector machine learning method with second-order cone programming for large-scale problems, Applied Intelligence 23(3) (2005) 219–239.**
9. **P. S. Bradley and O. L. Mangasarian, Massive data discrimination via linear support vector machines, Optimization Methods and Software 13(1) (2000) 1–10.**
10. **D. R. Musiant and O. L. Mangasarian, Large scale kernel regression via linear programming, Machine Learning 46(1–3) (2002) 255–269.**
11. **Suykens, J. A. K. Least squares support vector machine classifiers / J. A. K. Suykens, J. Vandewalle // Neural processing letters. – 1999. – vol. 9, Iss. 3. – pp. 293 – 300.**
12. **J. A. K. Suykens and J. Wandewalle, Least squares support vector machine classifiers, Neural Processing Letter 9(3) (1999) 293–300.**
13. **J. A. K. Suykens, T. Van Gestel, J. De Brabanter, B. D. Moor and J. Vandewalle, Least Squares Support Vector Machines (World Scientific, Singapore, 2002).**
14. **T. Van Gestel, J. A. K. Suykens, B. Baesens, S. Viaene, J. Vanthienen, G. Dedene, B. De Moor and J. Vandewalle, Benchmarking least squares support vector machine classifiers, Machine Learning 54(1) (2004) 5–32.**
15. **Guo, Y. M. Ran, C. B. Li, X. L. Ma, J. Z. Zhang, L. Weighted prediction method with multiple time series using multi-kernel least squares support vector regression, (2013) 188-194.**
16. **Jianping Li, Zhenyu Chen, Liweiqwi, Weixuan Xu, Feature Selection Via Least Squares Support Feature Machine (2014) 1-17.**
17. **Vapnik V. The nature of statistical learning theory. New York, USA Springer Verlag, 1995.**
18. **Vapnik V. Statistical learning theory. John Wiley and Sons, New York, 1998.**