УДК 004.023

ПРИМЕНЕНИЕ ЯДЕРНОГО ТРЮКА ДЛЯ УМЕНЬШЕНИЯ РАЗМЕРНОСТИ ПРИ БАЙЕСОВСКОЙ ОПТИМИЗАЦИИ

Антонов К.А. Университет ИТМО
Научный руководитель – к.т.н. Буздалова А.С.
Университет ИТМО

Алгоритмы байесовской оптимизации применяются для нахождения близких к оптимальным значений дорогих для вычисления функций. В работе предлагается модификация этого класса алгоритмов, позволяющая применять их к задачам с высокой размерностью. Модификация заключается в применении ядерного метода анализа главных компонент к задаче в процессе работы алгоритма.

Введение.

Задачи оптимизации, где вычисление целевой функции является ресурсоемким и для которых недоступна аналитическая информация, принято рассматривать как Black-Box задачи. В таких задачах наиболее естественный подход к оптимизации – это использование эвристик для нахождения достаточно хороших решений этих задач за относительно небольшое время, даже если эти решения не будут являться оптимальными. Действительно, во многих промышленных приложениях встречаются функции, вычисление которых требует много ресурсов, но при этом необходимо находить хорошие решения за ограниченное время. В таких приложениях оказываются полезными алгоритмы, основанные на построении суррогатных моделей, так как они позволяют получать просто-вычислимые приближения исходных сложных функций. Приближения используются, чтобы заменить прямую оптимизацию исходной функции на оптимизацию приближенной модельной функции и, таким образом, получить хороших кандидатов на решение исходной задачи. Тем не менее, алгоритмы, строящие вероятностные модели, обычно показывают хороший результат, когда размерность задачи относительно небольшая. В частности, байесовская оптимизация теряет свою эффективность, когда число измерений превышает 20, однако современные приложения требуют значительно более высокой размерности.

Основная часть.

Оптимизация функции классическим байесовским алгоритмом происходит в несколько итераций. На каждой итерации строится модель исходной функции с помощью которой ищется лучшая точка для вычисления значения функции. Далее в этой точке вычисляется значение исходной функции и на следующей итерации информация об этой точке добавляется в модель. Такой алгоритм используется в случае, когда количество измерений области определения функции для оптимизации относительно небольшое. Известно, что когда количество измерений превосходит 20, такой алгоритм находит достаточно хорошие решения медленнее алгоритмов, использующих эволюционные стратегии.

В работе предлагается уменьшить размерность задачи с помощью нелинейного метода анализа главных компонент. Этот метод уменьшения размерности задачи применяется в начале каждой итерации байесовской оптимизации, далее модель функции строится в пространстве меньшей размерности и далее для новой полученной точки производится обратное преобразование ее координат в исходное пространство.

Реализация такого алгоритма, помимо классической басовской оптимизации, использует следующие компоненты.

1)Метод для настройки параметров ядра, используемого для уменьшения размерности задачи. Такая настройка происходит на каждой итерации. Для ее осуществления решается

следующая задача оптимизации — выбор комбинации из ядра и параметров для него таким образом, чтобы как можно меньше главных компонент содержали 90 процентов от суммы дисперсий всех главных компонент.
2)Обратное преобразование координат точек. Такое обратное преобразование происходит после нахождения новой точки в пространстве меньшей размерности. Для реализации используется приближенный метод восстановления прообраза точки в предположении, что прообраз является линейной комбинацией других точек в исходном пространстве.
3)Перенос изначальных ограничений на множество, в котором ищутся решения, в пространство меньшей размерности. Такой перенос происходит при построении модели функции на каждой итерации. Для конкретных ядер перенос удается сделать с помощью анализа их структуры. В общем случае предлагается применять метод, основанный на сэмплировании большого количества точек в исходном пространстве без вычисления исходной функции в них, переноса этих точек в пространство меньшей размерности и вычисления радиуса шара, содержащего все эти точки.

Выводы.

Результатом этого исследования является алгоритм, который планируется применять в приложениях, требующих оптимизации сложных функций. Примеры таких приложений – это оптимизация топологии, управление роботами, разработка топологии VLSI.