Московский государственный университет имени М. В. Ломоносова Факультет
Вычислительной математики и кибернетики
Кафедра Математических методов прогнозирования

КУРСОВАЯ РАБОТА

Методы сравнения траекторий

Выполнил:
студент 317 группы
Кудрявцев Георгий Алексеевич

Научный руководитель:
д.ф-м.н., профессор
Дьяконов Александр Геннадьевич

Москва, 2015

Содержание

1 Введение **2**

2 Цели работы **2**

3 Существующие метрики **2**
3.1 Расстояние Хаусдорфа(HF) . . . 2
3.2 Модифицированное расстояние Хаусдорфа(MHF) . . . 2
3.3 Интерполяция на основе модифицированного расстояния Хаусдорфа(IMHF) 3
3.4 OWD (One Way Distance) . . . 3
3.5 Dynamic Time Warping . . . 3

4 Конкурс Driver Telematics Analysis **4**
4.1 Авторское решение . . . 4
4.1.1 Способ обучения . . . 4
4.1.2 Способ кросс-валидации . . . 5
4.1.3 Извлечение статистических признаков . . . 5

5 Инвариантные к поворотам методы сравнения траекторий **5**
5.1 Реализация метода статьи Rotation Invariant Distance Measures for Trajectories . . . 5
5.2 Реализация метода статьи Affine Invariant Dynamic Time Warping and its Application to Online Rotated Handwriting Recognition . . . 6
5.3 Генерация выборки и сравнение алгоритмов . . . 8

6 Решения других участников задачи DTA **9**

7 Выводы **9**

Список литературы **10**

1 Введение

С развитием Интернета географическая информационная система (ГИС) и услуги местоположения пользователя (Location-based service) играют важную роль в различных приложениях. Все больше и больше устройств способны собирать, обрабатывать и хранить информацию о местоположении подвижных объектов, которые представлены в форме траекторий.

Анализ сходства траекторий применяется в таких областях как экология, биология, телематика, геоинформатика.

В биологии и экологии сходства траекторий используются для наблюдений за миграцией животных [6]. Анализ маршрутов диких животных является важнейшим элементом их изучения, а так же хорошим показателем экологической обстановки в природе. Важной подзадачей в исследовании миграций является нахождение главных маршрутов. Траектории диких животных довольно хаотичны, поэтому для биологов важно определять главные пути, по которым перемещаются звери [5].

Также анализ сходства траекторий применяется в распознавании рукописных текстов. Символы можно представить в виде последовательности точек. Следовательно при помощи сравнения траекторий возможна их классификация и кластеризация. [1]

Анализ траекторий играют важнейшую роль в телематике. При помощи него выполняются такие задачи как оптимизация маршрутов, возможность обнаружения угнанного транспорта, а также навигация водителей в незнакомой местности.

Последние исследования в этой области посвящены нахождению различных патернов в стиле вождения водителей. [8, 9] В частности телематика представляет собой хороший способ оценки риска страхового случая водителя. При помощи анализа траекторий можно непосредственно оценивать поведение водителя на дороге. На платформе Kaggle прошел конкурс Driver Telematics Analysis [12], который как раз посвящен этой теме.

2 Цели работы

- Обзор методов сравнения траекторий.
- Решение задачи DTA [12] и обзор решений других участников.
- Обзор методов сравнения траекторий инвариантных к афинным преобразованиям.

3 Существующие метрики

3.1 Расстояние Хаусдорфа(HF)

Пусть есть два набора точек(траекторий) A и B. d – расстояние между двумя точками. Тогда H – расстояние Хаусдорфа между траекториями A и B:

$$\begin{aligned} H(A,B) &= \max(h(A,B), h(B,A)), \\ h(A,B) &= \max_{a_i \in A}(\min_{b_j \in B} d(a_i, b_j)), \\ h(B,A) &= \max_{b_i \in B}(\min_{a_j \in B} d(b_i, a_j)). \end{aligned} \tag{1}$$

3.2 Модифицированное расстояние Хаусдорфа(MHF)

Расстояние Хаусдорфа чувствительно к выбросам. Всего одна точка может сильно изменить ответ. Чтобы метрика стала менее чувствительна к выбросам, ее можно немного

подкорректировать.

$$h(A,B) = \frac{1}{|A|}\sum_{a_i \in A}(\min_{b_j \in B}\ d(a_i, b_j)) \tag{2}$$

3.3 Интерполяция на основе модифицированного расстояния Хаусдорфа(IMHF)

Чтобы уменьшить чувствительность расстояния между траекториями от точек измерения, можно проводить интерполяцию по точкам траекторий. В данном случае берется среднее арифметическое. Этот метод является одним из лучших основанных на метрике Хаусдорфа.

$$\begin{aligned} &h(A,B) = \frac{1}{|A|}\sum_{a_i \in A}(\min_{b_j, b_{j-1} \in B}\ d(a_i, \overline{b_j, b_{j-1}})), \\ &\text{где } \ \overline{b_j, b_{j-1}} = \frac{b_j + b_{j-1}}{2}. \end{aligned} \tag{3}$$

3.4 OWD (One Way Distance)

OWD является простым и эффективным способом вычисления расстояния между траекториями. Сначала задается расстояние между точкой p и всей траекторией Tr.

$$D_{point}(p, Tr) = \min_{q \in Tr}\ d(p, q) \tag{4}$$

Далее находим расстояние между траекториями Tr_1 и Tr_2 следующим образом:

$$\begin{aligned} D_{owd}(Tr_1, Tr_2) &= \frac{1}{|Tr_1|}\sum_{p \in Tr_1}(D_{point}(p, Tr_2) \\ D(Tr_1, Tr_2) &= \frac{1}{2}(D_{owd}(Tr_1, Tr_2) + D_{owd}(Tr_2, Tr_1) \end{aligned} \tag{5}$$

3.5 Dynamic Time Warping

Впервые этот метод был успешно применен в распознавании речи [7]. В настоящее время он активно используется в распознавании жестов, рукописных текстов, наблюдением за дикими животными. Рассмотрим две траектории

$$\begin{aligned} Q &= q_1, q_2, q_3 ... q_n \\ C &= c_1, c_2, c_3 ... c_m \end{aligned} \tag{6}$$

Сначала строится матрица расстояний D порядка $n \times m$, где $D_{i,j} = d(i,j)$ Затем строится матрица трансформаций K порядка $n \times m$, где

$$K_{i,j} = \begin{cases} D_{i,j} + min(K_{i-1,j}, K_{i-1,j}, K_{i-1,j-1}) & \text{, если i} > 1 \text{ и j} > 1 \\ D_{i,j} & \text{, если i} = 1 \text{ или j} = 1. \end{cases} \tag{7}$$

После заполнения матрицы деформации строится путь трансформации. Это последовательность элементов матрицы трансформации, которая минимизирует расстояние между траекториями.

Пусть p путь трансформации $p = (p_1, p_2, ...p_k)$, где $p_l = (q_i, c_j)$. Тогда он должен удовлетворять следующим требованиям.

Граничные условия: $p_1 = d(q_1, c_1)$ и $p_k = d(q_n, c_m)$. Это означает, что пусть трансформации начинается на начальных точках траекторий и кончается на конечных точках траекторий.

Непрерывность: Для $p_l = d(q_i, c_j)$ и $p_{l-1} = d(q_{i'}, c_{j'})$ выполняется $i - i' \leq 1$, $j - j' \leq 1$. Это значит, что путь трансформации состоит из смежных ячеек матрицы трансформации
Монотонность: Для $p_l = d(q_i, c_j)$ и $p_{l-1} = d(q_{i'}, c_{j'})$ выполняется $q_i - q_{i'} \geq 0$ и $c_j - c_{j'} \geq 0$. Это гарантирует, что путь трансформации не будет проходить через одну точку несколько раз.

Среди всех возможных путей трансформации, которые удовлетворяют условиям, выбирается минимальный. DTW расстояние между двумя последовательностями через оптимальный путь трансформации выражается следующим образом.

$$D_{DTW}(Q, C) = \frac{p_k}{k} \tag{8}$$

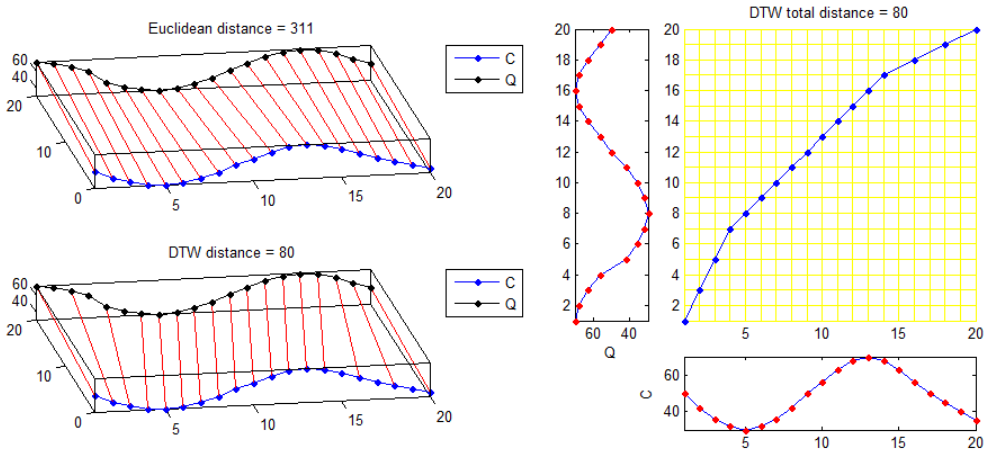

Рис. 1: Dynamic Time Warping

На рис.1 изображена работа DTW на двух одинаковых траекториях, у которых помечены разные точки. Слева красными отрезками соединены точки, между которыми вычисляется расстояние. Справа изображена матрица трансформации.

4 Конкурс Driver Telematics Analysis

На платформе Kaggle проводился конкурс Driver Telematics Analysis [12]. Организаторы конкурса предложили следующую задачу. Даны 2736 папки. В каждой папке находится 200 траекторий. Организаторы утверждают, что большинство траекторий в каждой папке соответствует движению одного водителя(для различных папок различные водители). Предлагается выяснить, какие траектории принадлежат ему, а какие нет.

Также организаторы делали преобразование над траекториями. Они поворачивали их на случайные углы. Начало и конец вырезали. После этого траектории смещали так, чтобы их начальная точка была равна (0, 0).

Требуется определить какие траектории принадлежат данному водителю, а какие нет. Критерием качества на лидерборде выступал AUC [10].

4.1 Авторское решение

4.1.1 Способ обучения

Будем использовать подход обучения с учителем. Для этого построим тренировочную выборку следующим образом: в качестве положительных примеров берем 200 траекторий данного водителя, а в качестве отрицательных – 50 случайно выбранных траекторий других водителей. Тестовая выборка будет состоять из первых 200 траекторий.

По условию задачи положительных примеров больше, чем отрицательных. Поэтому в тренировочной выборке создается дисбаланс в сторону положительных примеров, чтобы алгоритм машинного обучения чаще предсказывал их.

Только 50 отрицательных примеров было использовано при создании тренировочной выборки. Но в задаче их намного больше. Чтобы учитывать большее количество отрицательных примеров, процесс создания выборки и получения ответа проводится несколько раз. Окончательный результат равен среднему значению ответов.

4.1.2 Способ кросс-валидации

Построим тренировочную выборку следующим образом: в качестве положительных примеров берем 200 траекторий данного водителя и 50 случайно выбранных траекторий других водителей, а за отрицательные – 250 случайно выбранных траекторий других водителей. Тестовая выборка будет состоять из первых 250 траекторий, причем первые 200 являются положительными примерами, а остальные 50 – отрицательными.

Чтобы учитывать большее количество отрицательных примеров, процесс создания тренировочной выборки для кросс-валидации и вычисление качества работы алгоритма проводится несколько раз. Итоговая оценка равна среднему значению качества.

4.1.3 Извлечение статистических признаков

Были использованы следующие статистические признаки: среднее значение, дисперсия, максимальное значение скорости и ускорения, время простоя и максимальное время разгона, отношение дисперсии скорости к максимальной скорости. Аналогичные признаки для изменения угла направления относительно предыдущей точки и относительно центра масс. Были проведены попытки использования гистограмм скоростей и ускорений, но они почему-то ухудшали результат. Обучение проводилось при помощи Random Forest, GBM, Logistic Regression из библиотеки scikit-learn.

Алгоритм	Результат на лидерборде
Random Forest	0.86116
GBM	0.84279
Logistic Regression	0.74810

Таблица 1: Точность разных алгоритмов на лидерборде

Лучший результат показал Random Forest при параметрах max_features=9 и n_estimators=250.

5 Инвариантные к поворотам методы сравнения траекторий

5.1 Реализация метода статьи Rotation Invariant Distance Measures for Trajectories

Этот метод был создан для сравнения рукописных чисел [1]. Основная идея – перевод траектории в систему координат инвариантной к поворотам . Пусть $P = [P_1, ..P_n]$ – траектория.

$V_t = P_t - P_{t-1}$

$V_{ref} = P_{t-1} - P_{cMass}$, где P_{cMass} – центр масс траектории.

$\alpha_t = \text{sign}(V_t, V_{ref}) \cdot \arccos(\frac{<V_t V_{ref}>}{\|V_t\|\|V_{ref}\|})$, где функция sign определена следующим образом

$$\text{sign}(V_t, V_{ref}) = \begin{cases} 1 & \text{, если } [V_t \times V_{ref}] \cdot [0\ 0\ 1]^T > 0 \\ -1 & \text{, если } [V_t \times V_{ref}] \cdot [0\ 0\ 1]^T \leq 0 \end{cases}$$

Полученное α_t есть изначальная траектория в новой системе координат. Далее в новой системе координат надо вычислить расстояние между траекториями при помощи DTW.

Основным минусом этой метрики – инвариантность к масштабу. Поэтому вытянутые траектории слабо различимы с малым количеством поворотов.

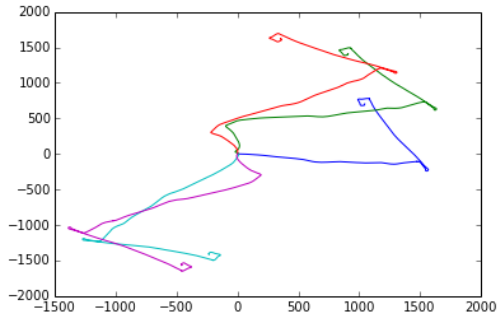

Рис. 2: Хорошая работа алгоритма

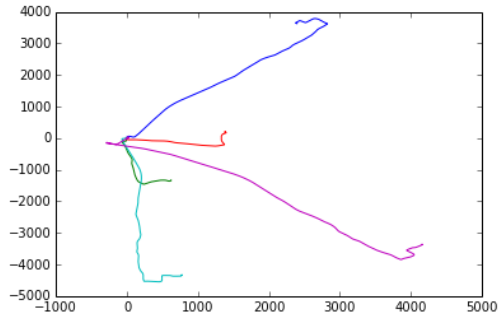

Рис. 3: Плохая работа алгоритма

На рис.2 и на рис.3 изображены траектории, которые были выделены при помощи вышеупомянутой метрики. Видно, что на траекториях с большим количеством поворотов метод работает хорошо, но если поворотов мало, то метрика работает неправильно.

5.2 Реализация метода статьи Affine Invariant Dynamic Time Warping and its Application to Online Rotated Handwriting Recognition

Этот метод был также создан для сравнения рукописных чисел [2]. Основная идея – использовать EM алгоритм для нахождения матрицы афинного преобразования, с помощью которой можно повернуть одну из траекторий для их наилучшего приближения. Пусть даны две последовательности одинаковой длины. $R = [r_1, r_2, ..., r_n]$
$T = [t_1, t_2, ..., t_n]$ Надо минимизировать следующий функционал:

$$\sum_{i=1}^{k} \|t_i - r_i A\|, \text{где A - матрица поворота.} \tag{9}$$

Т.е. надо найти такую матрицу поворота, которая лучше всех сближала две траектории. Эту задачу можно решить простым способом. Пусть $D_r = [r_1, r_2, ...r_k]^T$
$D_t = [t_1, t_2, ...t_k]^T$

Тогда матрицу поворота A можно найти при помощи равенства нулю производной функционала (9). $A = (D_r^T D_r)^{-1} D_r^T D_t$
Но при большом размере траекторий эта задача становится трудоемкой. Поиск более быстрого решения является ортогональной проблемой Прокруста. Эта задача хорошо решается при помощи алгоритма Кабша [4].

Но мы имеем траектории разной длины в общем случае. Эту проблему авторы статьи решили следующим образом. Они применили EM-алгоритм, в котором на E-шаге минимизируется функционал по пути трансформации, а на M - шаге вычисляется новый путь трансформации.

Входные данные: Траектории T и R.
Результат: $A^{(k)}$.

1 k=1;
2 $w^{(0)} = DTW\_PATH(T, R)$;
3 **до тех пор, пока** *не сошлось* **выполнять**
4 $\quad A^{(k)} = \text{argmin}_A \sum_{i=1}^{k} \| t_{w_i^{(k-1)}} - r_{w_i^{(k-1)}} A \|$;
5 $\quad w^{(k)} = DTW\_PATH(T, A^{(k)} \cdot R)$;
6 $\quad k = k + 1$;
7 **конец цикла**

Алгоритм 1: EM алгоритм

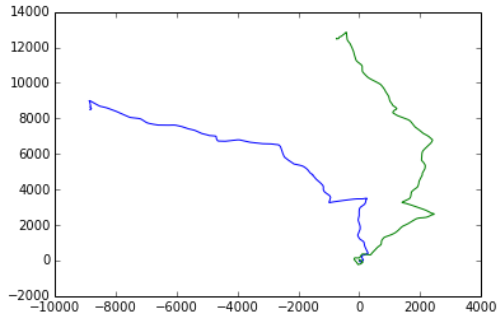

Рис. 4: Изначальное положение траекторий

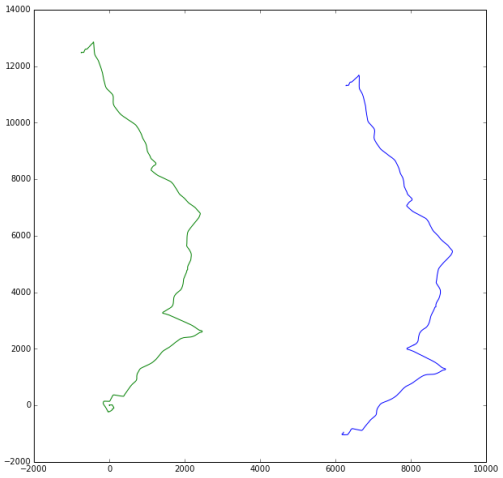

Рис. 5: Первый проход EM алгоритма

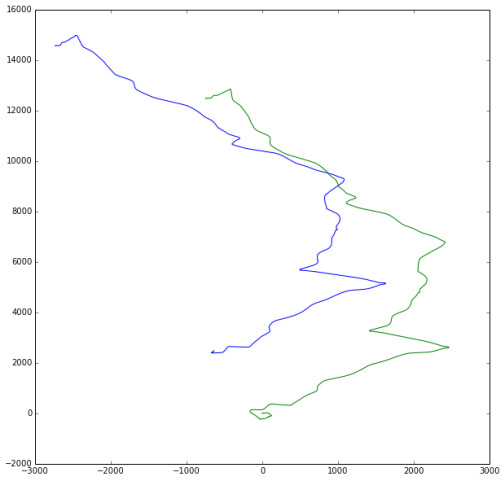

Рис. 6: Четвертый проход EM алгоритма

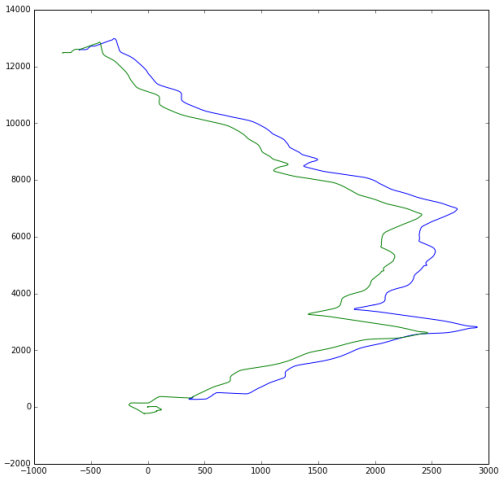

Рис. 7: Десятый проход EM алгоритма

На рис.4 - 7 показана работа EM-алгоритма на различном количестве итераций.

5.3 Генерация выборки и сравнение алгоритмов

Проверять качество метрик непосредственно на платформе Kaggle было неправильно, т.к. в конкурсе надо было находить траектории принадлежавшие одному водителю, а траектории могли принадлежать одному водителю и одновременно абсолютно непохожими друг на друга.

Сравнение алгоритмов произведено на искусственной выборке. Одна половина выборки состояла из шума, т.е траекторий разных водителей. Другая состояла из искусственных кластеров. Кластеры были созданы так же, как это делали организаторы конкурса. Берем траекторию, поворачиваем на случайный угол, с начала и конца удаляем случайную по длине(но не больше, чем 5% от длины траектории) часть траектории. Для каждой траектории этот процесс повторяем несколько раз для создания кластера.

Далее из траекторий удаляем лишние точки. Это очень важно для быстрой скорости работы. Строим матрицу расстояний. Далее этой матрице выполняем кластеризацию при помощи метода DBscan. Качество кластеризации оцениваем при помощи метрики Adjusted Rand index [11].

Сравнение алгоритмов было произведено на выборках с различным количеством кластеров и шума. Один кластер состоит из 30 траекторий. Количество шума равно числу всех траекторий в кластерах.

На рис.8 видно, что в среднем второй метод работает лучше, чем первый . Это связано с тем, что второй метод не является инвариантным к масштабированию.

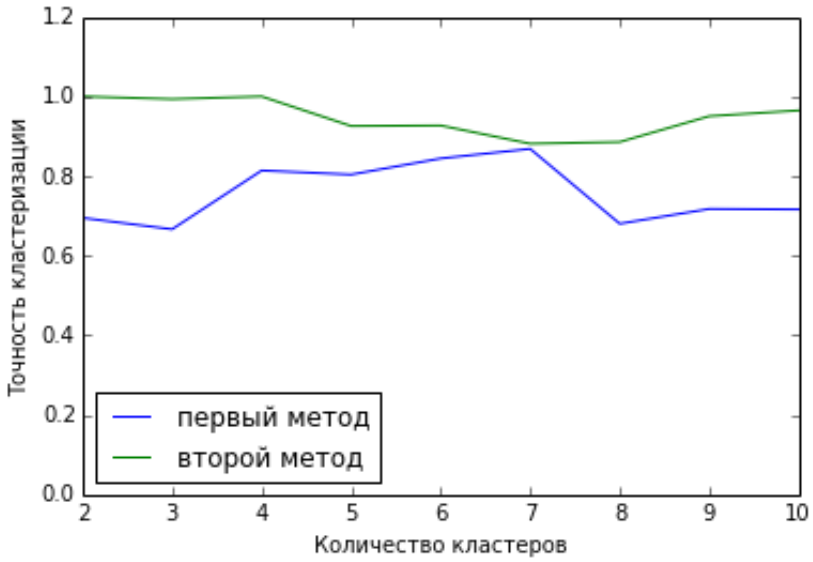

Рис. 8: Сравнение первого и второго методов

6 Решения других участников задачи DTA

Для проблемы Прокруста необходимы траектории одинаковой длины. Другие участники конкурса обошли эту проблему иным образом. Они сравнивали части траекторий одинаковой длины между собой. В этой задаче эта метрика подходит лучше, так как было много траекторий, которые являлись частью другой. Мои представленные метрики это не отлавливали.

Еще один способ решения проблемы различной длины траекторий следующий. Координаты переведены в новую систему, где точки зависят не от времени, а от длины. Длина нормированная: от 0 до 1. Ее разбивали на одинаковое количество отрезков равной длины. В этом случае у траекторий будет одинаковое количество точек. Если не было точки соответствующей какому-нибудь значению нормированной длины, то ее получали при помощи линейной комбинации соседних. Далее вычислялась кривизна траектории. Итоговой метрикой являлась кросс-корреляция [13] двух векторов кривизны.

Другое решение было такое. Строилась гистограмма углов поворотов водителей на каждой траектории. Углы брались по модулю, т.е. не имело значение в какую сторону производился поворот. Затем данные гистограммы сравнивались следующим образом. Пусть $A = [a_1, a_2, ..., a_n]$ и $B = [b_1, b_2, ..., b_n]$

$$D(A, B) = \frac{\sum_{i=1}^{n} |a_i - b_i|}{\sum_{i=1}^{n} a_i + b_i} \tag{10}$$

Если близких траекторий оказалось n, то ответ статистической модели увеличивался на $n \cdot k$, где k - настраиваемая константа.

7 Выводы

Среди неинвариантных к афинным преобразованиям методов сравнения траекторий самым лучшим является DTW.

В задаче DTA организаторы хотели, чтобы участники конкурса решали задачу, опираясь только на статистические признаки, и не смотрели на геометрию траекторий. Для этого они сделали множество всевозможных преобразований(повороты, удаление начала и конца пути). В топ лидерборда попало много решений, которые основывались только на статистических данных. Но самые лучшие решения использовали еще и степень сходства траекторий. Даже такие испорченные траектории несут достаточно информации для их правильного сравнения.

Самыми лучшими статистическими признаками являются гистограммы и квантили скоростей и ускорений.

Среди представленных неинвариантных к афинным преобразованием методов метрика, основанная на EM-алгоритме, показала лучший результат.

Список литературы

[1] Yu Qiao, Makoto Yasuhara. Affine Invariant Dynamic Time Warping and its Application to Online Rotated Handwriting Recognition. 18th International Conference on Pattern Recognition, 2006

[2] Michail Vlachos, Gautam Das. Rotation Invariant Distance Measures for Trajectories. Tenth ACM SIGKDD international conference on Knowledge discovery and data mining, 2004.

[3] Peng Chen, Junzhong Gu, cDehui Zhu, Fei Shao A Dynamic Time Warping based Algorithm for Trajectory Matching in LBS. International Journal of Database Theory and Application Vol. 6, No. 3, June, 2013

[4] Kabsch, Wolfgang. A solution for the best rotation to relate two sets of vectors. Acta Cryst, 1976.

[5] Eric Palm, Scott Newman2, Diann Prosser, Mapping migratory flyways in Asia using dynamic Brownian bridge movement models. Movement Ecology, 2015.

[6] Andrew Markham. On a Wildlife Tracking and Telemetry System: A Wireless Network Approach, 2008.

[7] L. Rabiner, B.H. Juang. Fundamentals of Speech Recognition. Prentice Hall PTR, 1993

[8] Kexin Nie, Luyan Wu, Jiafan Yu. Driving Behavior Improvement and Driver Recognition Based on Real-Time Driving Information. CS229 Project, 2013.

[9] Anastasia Bolovinou, Angelos Amditis. Driving Style Recognition for Co-operative Driving: A Survey. The Sixth International Conference on Adaptive and Self-Adaptive Systems and Applications, 2014

[10] Area under the curve. `http://en.wikipedia.org/wiki/Receiver_operating_characteristic`

[11] Adjusted Rand index. `http://en.wikipedia.org/wiki/Rand_index`

[12] Kaggle. Driver Telematics Analysis. `http://www.kaggle.com/c/axa-driver-telematics-analysis`

[13] Cross-correlation `http://en.wikipedia.org/wiki/Cross-correlation`