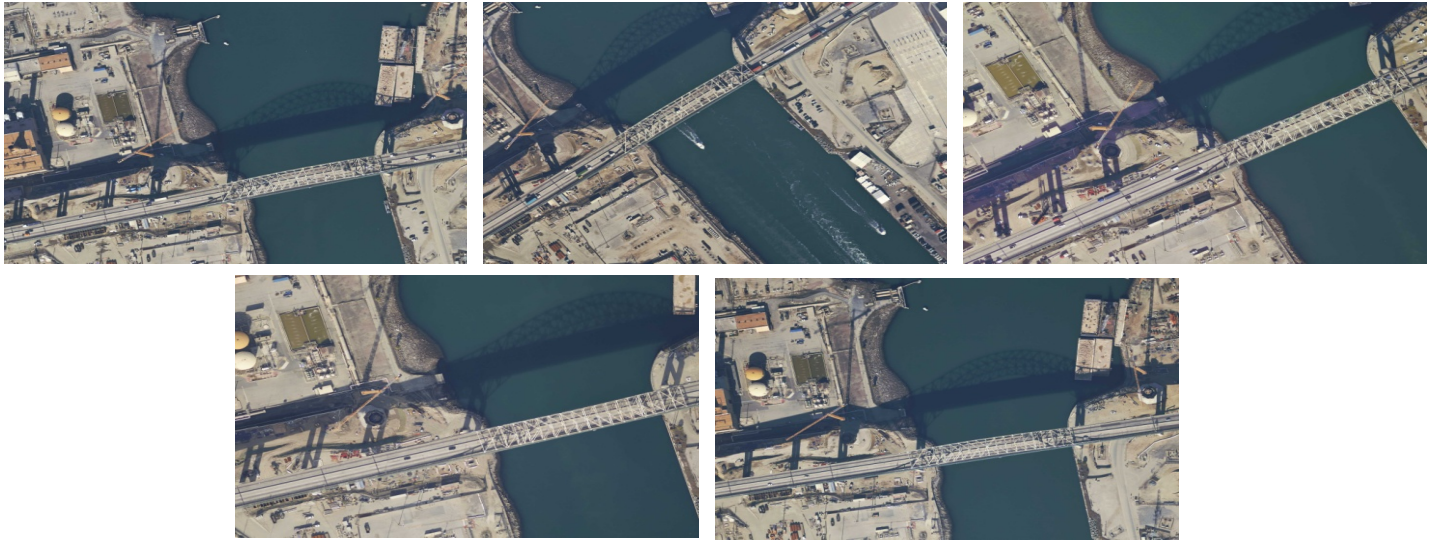
Kaggle Draper Satellite Image Chronology

Участники:
Шилин Сергей
Блянкинштейн Наташа
Найденов Алексей

Соревнование: исходные данные, правила и задача

Исходные данные:
Датасет содержит 1720 аэроснимков. Все снимки разбиты на 344 сета (по 5 снимков).
Каждый сет содержит 5 снимков одной местности за 5 дней (каждый снимок сделан в определенный день).
Каждый снимок представляет собой изображение в формате JPEG (со сжатием) и TIFF (без сжатия) с разрешением 3099 на 2329 пикселей



Соревнование: оценка качества

Public leaderboard: 17%

Коэффициент корреляции Спирмена:

$$\rho = 1 - \frac{6}{n(n-1)(n+1)} \sum_{i=1}^{n} (R_i - S_i)^2$$

Свойства :

- Важен порядок элементов в списке.
- 1 — прямой порядок, -1 — обратный порядок.
- При accuracy 0.6 spearmanr может быть как 0.9, так и -0.6 в зависимости от порядка элементов списка

Соревнование: исходные данные, правила и задача

Исходные данные:

1. Обучающая выборка –70 сетов (350 снимков). ID снимка соответствует хронологическому порядку.
2. Тестовая выборка – 274 сета (1370 снимков). ID снимка НЕ соответствует хронологическому порядку (порядок перепутан).

Задача:
Восстановить хронологический порядок в сетах тестовой выборки.

Инструменты :

1. MatLab,
2. Python: numpy, pandas, sklearn.

Can you put order to space and time?

Imagine a world where we can use satellite images to help find better access to clean water, prevent poaching of wildlife, predict storms more efficiently, optimize traffic patterns more readily, and inform human behaviors to mitigate the spread of disease.

Thanks to a marked increase of satellites in orbit, we will be able to capture images – and the information contained within – of nearly every place on Earth, every day by 2017. However, our ability to analyze datasets of these images has not advanced as quickly. Changes from day to day in images of the same location are subtle, can be hard to detect, and are difficult to understand in terms of their significance.

In this competition, Draper provides a unique dataset of images taken at the same locations over 5 days. Kagglers are challenged to predict the chronological order of the photos taken at each location. Accurately doing so could uncover approaches that have a global impact on commerce, science, and humanitarian works.

Данные: наблюдения, изучение природы данных

Проблема 1: Как измерить степень сходства/различия между снимками?
Рассмотренные решения:
1. Различия между относительными углами поворота и масштабами снимков
2. Среднеквадратичная ошибка яркости пикселей на пересекающейся области снимков
3. Отношения между статистиками яркости пикселей (выборочное среднее, медиана), между гистограммами яркости

Проблема 2: Как определить хронологический порядок?
Косвенные признаки:
1. Краткосрочные явления – автомобили на парковках, тени, лужи, погодные явления
2. Долгосрочные явления – стройки домов, посевы на полях
3. Характерные траектории полета в различные дни
4. Тени

Данные: Feature Engineering

Идея 1. Анализ матрицы трансформации между 2 снимками

Источник вдохновения: Урок «Find Image Rotation and Scale Using Automated Feature Matching» от MathWorks.

План:

1. Найти расположение и описание контрольных точек на паре снимков (SURF);
2. Выделить значимые контрольные точки на снимках;
3. Найти соответствие между контрольными точками пары снимков (RANSAC), получить матрицу аффинного преобразования второго снимка в систему координат первого снимка;
4. Выделить свойства из матрицы преобразования: сдвиг по горизонтали, сдвиг по вертикали, угол поворота, масштаб;
5. Агрегировать свойства, выделенные из матриц преобразования относительно всех четырех остальных снимков

Данные: Feature Engineering

Отобранные признаки:

1. Индекс снимка в списке, упорядоченном по углу поворота = число снимков с отрицательным углом относительно данного (категориальное)
2. Индекс снимка в списке, упорядоченном по масштабу = число снимков с масштабом < 1 относительно данного (категориальное)
3. Среднее значение относительного масштаба (непрерывное)
4. Наибольшее абсолютное значение относительного угла (непрерывное)
5. Наименьшее абсолютное значение относительного угла (непрерывное)
6. Наименьшее абсолютное значение относительного угла при scale ~ 1(непрерывное)

Scale					
FrameId	**1**	**2**	**3**	**4**	**5**
1	1,000	1,822	1,060	0,998	1,040
2	3,269	1,000	1,043	0,974	1,021
3	0,944	0,958	1,000	0,940	0,980
4	1,001	1,019	1,062	1,000	1,041
5	0,962	0,983	1,020	0,961	1,000

Angle					
FrameId	**1**	**2**	**3**	**4**	**5**
1	0,000	65,388	-4,922	-6,666	-15,808
2	73,572	0,000	11,000	8,792	0,041
3	4,728	-11,006	0,000	-2,093	-10,921
4	6,670	-8,893	1,833	0,000	-8,878
5	15,775	0,004	10,904	8,878	0,000

Данные: модели, ансамбли

Рассмотренные модели:

1. Multiclass Logistical Regression
2. ExtraTrees Classifier
3. Multiclass SVM
4. Ансамбль этих моделей

Валидация моделей:

5. Кросс-валидация на обучающей выборке (0.88 accuracy)
6. Отклонение кол-ва снимков, попавших в каждый класс, от (N/5) на тестовой выборке (СКО)

Лучшая модель, полученная на основании Идеи 1 – Multiclass SVM (Class-by-class classification fixed by OVR multiclass SVM)

Оценка модели:

– public score: 0.86686
– private score : 0.86545

Данные: Внимательное изучение

Идея 2. Кластеризация снимков

Мысли:

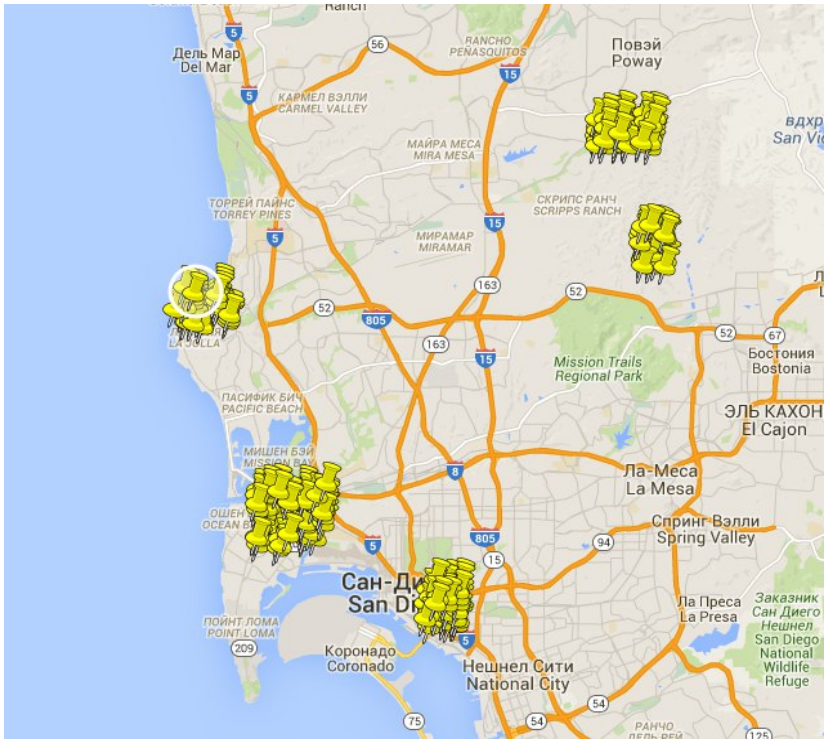
1. Данные на трейне и на тесте очень сильно различаются. Поэтому почти все модели, которые показывали хорошее (> 0.90) качество на валидации, были провальными на лидерборде
2. Стало понятно, что на основе этих фич трудно получить результат сильно лучше. Полезли смотреть на данные.

Наблюдения:

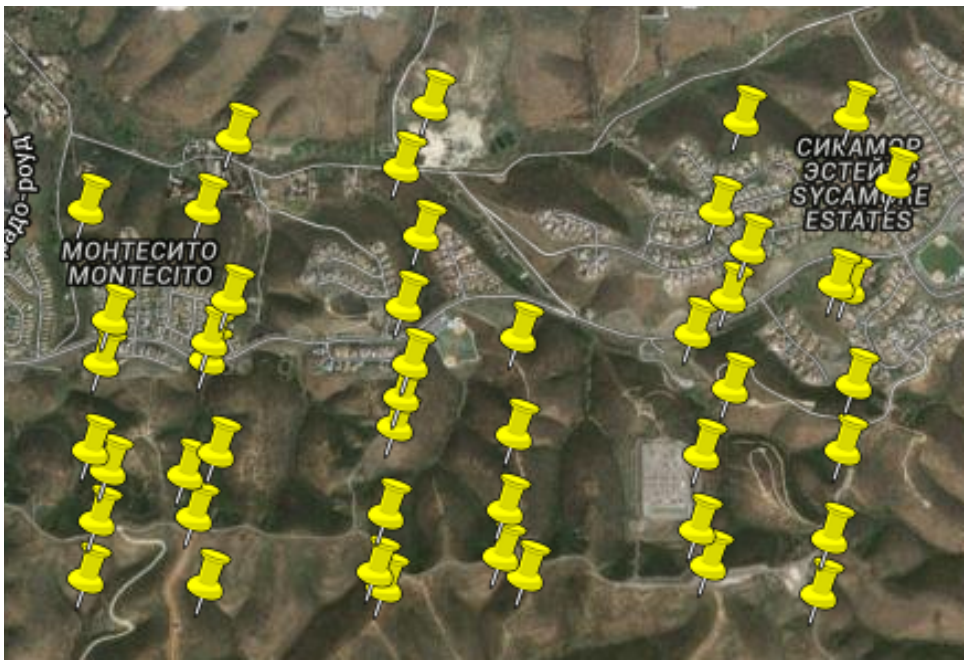
1. В нескольких сетах снято одно и то же или совсем похожее (соседнее) место
2. Многие сеты пересекаются, их можно собрать в пазлы
3. Наконец открыли форум. Добрыми участниками были размечены многие места на карте.
4. Вся выборка объединяется в 7 кластеров по местоположению (lat, lng)

Данные: Внимательное изучение

Кластеры в районе Сан-Диего



Кластеры группы С



Данные: Внимательное изучение

Идея 2. Кластеризация снимков

Мысли:

1. В Сан-Диего только тестовые сеты, все сеты из трейна расположены в Лос-Анджелес
2. Для каждого кластера можно собрать 5 пазлов, каждый из которых соответствует одному дню
3. Каждый кластер делится на несколько маленьких подкластеров, один подкластер соответствует одному пролёту самолета по прямой линии

Данные: Внимательное изучение

Идея 2. Кластеризация снимков

Дальнейшие действия:

1. Оставалось упорядочить 5 больших пазлов. Для нескольких сетов из одного кластера определялся порядок по высыхающим лужам, автомобилям, стройкам. По этим «эталонам» восстанавливался порядок в остальных сетах.
2. Порядок остальных сетов пытались восстанавливать автоматически по относительным углам между снимками в соседние дни. Оказалось, что на многих сетах масштаб и угол поворота снимков были почти одинаковы. Это давало большую ошибку. Нужны более тонкие различия.

Наблюдения:

В один день все снимки производились примерно в одно и то же время и при одних и тех же погодных условиях; оказалось, что тени изо дня в день меняются одинаково во всех сетах (внутри кластера, и даже между кластерами)

Данные: Внимательное изучение

Идея 2. Кластеризация снимков

Проблема:

В отличие от остальных кластеров, кластер E не поддавался такой разметке

Решение:

1. В качестве проверки сравнили по spearmanr полученный сабмит с сабмитом 0.86 для каждого сета; т. к. 0.86 — это достаточно большая корреляция, то снимки шли от первого к последнему примерно правильно.
2. На одном кластере из группы E, а именно E1, корреляция с решением 0.86 была около 0.9, с остальными ~ 0.2
3. Перепроверили остальные кластеры группы E, подогнали ответы таким образом, чтобы корреляция на всех сетах одновременно была большой.

Итоги

1. До победы не хватило внимательности и времени + на момент нашего вступления в соревнование уже было 3 лидера с 1.0000 на public.
2. Важно: в соревновании был разрешен hand labeling и human detection.
3. Решали соревнование ~ 1 мес, 0.86 получили через 2 недели, идея такой ручной разметки пришла за неделю.
4. Суммарно сделали 19 сабмитов
5. Результат: 8th/401

Ссылки:

https://www.kaggle.com/c/draper-satellite-image-chronology/
https://github.com/sergeyshilin/kaggle/tree/master/draper-satellite/