Московский Государственный университет имени Ломоносова
Факультет вычислительной математики и кибернетики
Кафедра математических методов прогнозирования

Вихрева Мария Викторовна

Распространение эпидемий в графах

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

Научный руководитель

д. ф.–м. н., профессор ММП
А. Г. Дьяконов

Москва, 2016

Содержание

1 Введение

В данной работе рассматривается задача предсказания распространения эпидемии в графе, где вершине соответсвует человек, ребру — наличие социальной связи (например, отношение дружбы). В последние годы появилось большое количество социальных сетей, и оказались доступны с ними связанные массивы данных. Такие данные могут обладать несколькими слоями и размерностями: есть как пользователи, так и связи между ними (возможно направленные), пользователь обладает как статичными свойствами, так и изменяющимися во времени. Идеей было реализовать метод, который бы использовал весь спектр доступной информации (использование профиля пользователя ранее не применялось).

Под «эпидемией» подразумевается не только болезнь в общепринятом смысле, но и любое распространяющееся свойство, присущее вершинам графа. Взяты также не любые «болезни», а лишь те, которые не подвержены «выздоровлению», то есть множество «заболевших» только расширяется со временем (далее не будем писать болезнь в ковычках).

В частности, рассмотрена задача прогнозирования распространения новости о Бозоне Хиггса по социальной сети Твиттер: предсказать вершины социального графа Твиттер, соответствующие пользователям, которые запостят новость в долговременной перспективе, если известна история сообщений предыдущих дней.

Документ разбивается на теоретическую и практическую части: разделы 2-5 содержат постановку задачи, обзорную информацию по необходимым методам и описание предлагаемого подхода к решению; в 6-ом приведены эксперименты, в 7-ом описаны результаты.

Раздел 2 включает формальную постановку задачи.

В разделе 3 приведен обзор классических подходов в эпидемиологии к задаче прогнозирования заражений. В ходе экспериментов была реализована стохастическая SI-модели на графе, и в разделе 6 приведено её сравнение с подходами ниже.

В качестве предлагаемого подхода решения задачи в разделе 5 приведена модель машинного обучения и процесс её обучения. Были протестированы модели логистической регрессии, персептрона, наивного байеса, случайных лесов.

В качестве стратегии получения предсказаний выбрана рекурсивная одношаговая модель, опирающаяся на наблюдения двух предыдущих шагов. В разделе 4 приведен обзор существующих других стратегий прогнозирования временных рядов, а в разделе 6 сравнение стратегий с рамках датасета Твиттер-сообщений.

Данная задача прогнозирования на таких данных решалась впервые, так как более стандартной потребностью в данной области является прогнозирование только количества постов (как например, в статье [1]). Лучшей моделью машинного обучения достигнуты 100% точность и 96% полнота на 4 днях сообщений с историей в 2,5 дня, что лучше по качеству всех приведенных в работе аналогов.

2 Постановка задачи

Рассмотрим граф $G = (V, E)$, в котором вершинам сопоставлены люди, а ребрам – наличие социальной связи. Такого рода графы часто называют социальными сетями. В момент времени t (t — дискретная величина) для графа определены:

- $S(t)$ — множество здоровых людей *(susceptible)*;
- $I(t) = E \setminus S(t)$ — множество больных людей *(infected)*;

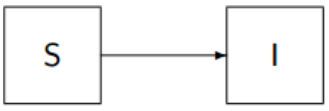


Рис. 1: Схема SI-модели.

Считается, что человек не может перейти из множества заболевших в множество здоровых, и выполнено условие $I(t) \subseteq I(t+1)$ для любого t. Также для простоты полагается, что ребра и вершины статичны и не изменяются со временем.

Цель — предсказание появления новых заболевших к моментам времени $t_0 + 1, t_0 + 2, \ldots, t_0 + n$ на основе истории заболеваний за t_0 предшествующих моментов времени. Рассматривается как задача предсказания количества новых заболевших $|I(t+1) \setminus I(t)|$, так и задача определения вершин из множества $I(t+1) \setminus I(t)$ в социальном графе, подвергшихся заболеванию в $(t+1)$-ый момент времени.

Описанная выше эпидемиологическая модель также называется SI-моделью, так как допускаются только перемещения *susceptible* $\rightarrow$ *infected*. Также существуют модели SIS, SIR *(R — recovered)* и другие. Для исследования была выбрана именно SI модель, так как она является наиболее простой и больше подходит для проверки применимости предлагаемого в работе подхода.

3 Обзор эпидемиологических подходов

3.1 Классическая модель SI

Данный подход основывается на утверждении: для любого промежутка времени верно, что количество человек, присоединившихся к больным, равно количеству человек, переставших быть здоровыми. Пусть

- β — вероятность заражения здорового при контакте с больным,
- за $n_s(t) = |S(t)|$ обозначим количество здоровых человек в момент времени t,
- за $n_i(t) = |I(t)|$ — количество больных.

Тогда опишем процесс распространения болезни дифференциальными уравнениями:

$$\begin{aligned} \frac{dn_s(t)}{dt} &= -\beta n_s(t) n_i(t); \\ \frac{dn_i(t)}{dt} &= \beta n_s(t) n_i(t). \end{aligned} \tag{1}$$

Используя $n_i + n_s = N = const$, перепишем:

$$\begin{aligned} \frac{dn_s(t)}{dt} &= -\beta n_s(t)\big(N - n_s(t)\big); \\ \frac{dn_i(t)}{dt} &= \beta\big(N - n_i(t)\big) n_i(t). \end{aligned} \tag{2}$$

Здесь произведение $n_s(t)\big(N - n_s(t)\big) = n_i(t)\big(N - n_i(t)\big) = n_s(t)n_i(t)$ равно количеству контактов за единичный промежуток времени в случае, если каждый здоровый контактирует с каждым больным. В первой строке расписано количество ушедших из множества здоровых (левая часть уравнения) как доля заразившихся при контактах (правая часть уравнения). Во втором уравнении количество новоприбывших в множество больных (левая часть уравнения) расписывается как доля заразившихся при контактах (правая часть уравнения).

Примеры решения системы уравнений для непрерывного и дискретного времени можно найти в работах [2], [3]. Характерная динамика количества здоровых и больных отображена на рис. 2.

Основным недостатком подхода является предположение о том, что больные и здоровые равномерно распределены по пространству. Как следствие, β является константой для всех участников социальной сети, и считается, что контакты происходят между всеми парами (*больной*, *здоровый*). Также модель предсказывает на уровне численности и не делает никаких предположений о том, кто именно заболевает в следующий момент времени.

3.2 Модель SI для графа «Мир тесен»

Систему уравнений выше можно переписать для случая, когда в каждый момент времени здоровый имеет примерно $\langle k \rangle$ контактов с больными, а значит $\langle k \rangle$ возможностей заразиться:

$$\begin{aligned} \frac{dn_s(t)}{dt} &= -\beta\langle k \rangle n_s(t)\big(N - n_s(t)\big); \\ \frac{dn_i(t)}{dt} &= \beta\langle k \rangle n_i(t)\big(N - n_i(t)\big). \end{aligned} \tag{3}$$

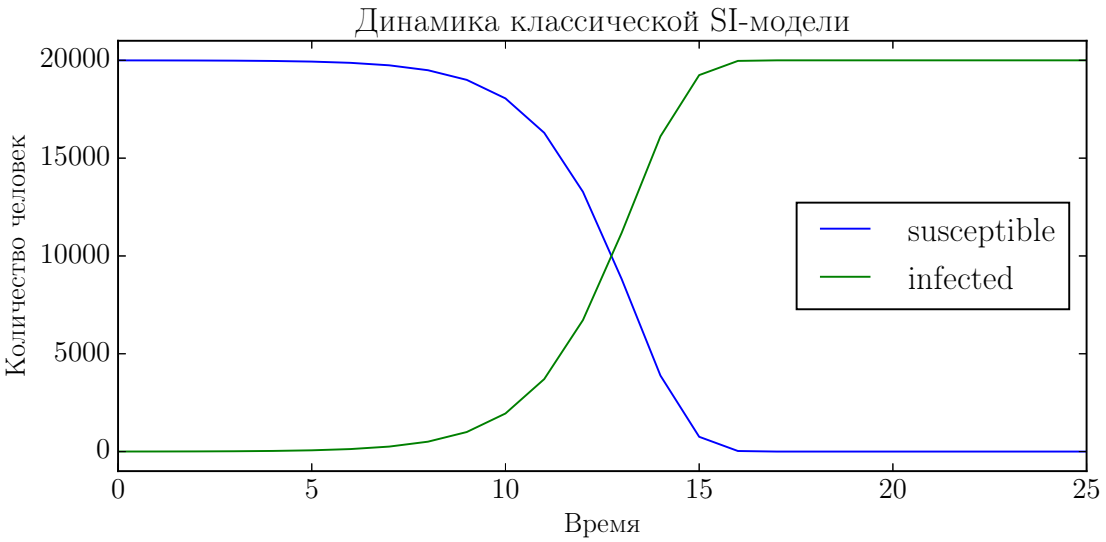


Рис. 2: Численное моделирование зависимости количества здоровых и больных от времени, $\beta = 0.00005$, $N = n_s(t) + n_i(t) = 20000$.

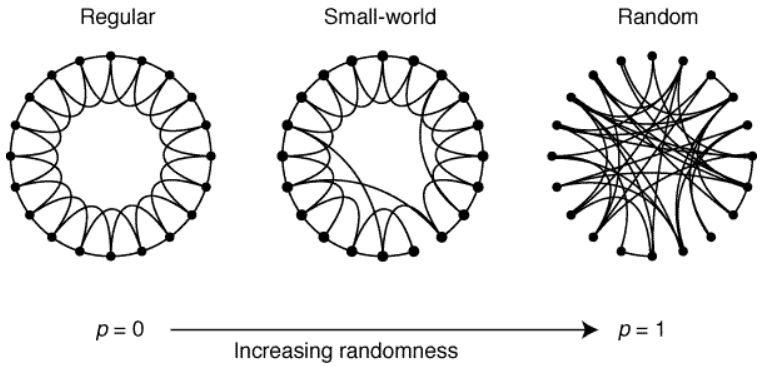


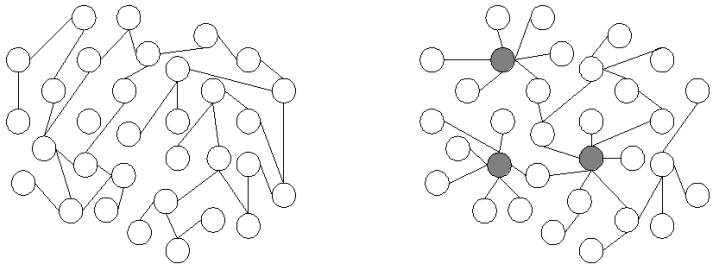
Рис. 3: Генерация графа «Мир тесен» по схеме Ватца-Строгаца (рисунок из [4]).

Примечательно, что данное расширение модели имеет интерпретацию для графа. А именно, болезнь распространяется так, как если бы люди были вершинами графа «Мир тесен», связи – его ребрами, а β — вероятность заразиться здоровому при контакте с больным в единичный промежуток времени.

Граф «Мир тесен» («Маленький мир») может быть смоделирован следующим способом (рис. 3): изначально берется кольцо из N вершин, где каждая вершина симметрично смежна с $2K$ ближайшими соседями; далее для каждой вершины каждое её ребро с вероятностью p обрывают на другом конце и соединяют с случайной вершиной. Процедура выдает случайный граф, в котором типичное расстояние L между двумя произвольно выбранными вершиными растет пропорционально логарифму от числа вершин N: $L \propto \log N$, а также в котором среднее количество соседей $\langle k \rangle = 2K$ [4].

Такие графы обладают приятным свойством: если взять две произвольные вершины a и b, то они с большой вероятностью не являются смежными, однако одна достижима из другой посредством небольшого количества переходов через другие вершины. Данное свойство также известно как «правило шести рукопожатий», согласно которому любые два человека на Земле разделены не более чем пятью цепочками общих знакомых. Примерами графов «Мир тесен» из жизни являются граф телефонных звонков, сеть интернет, сеть, где вершины — гены, связи — взаимное влияние генов.

Для графов «Мир тесен» верно, что каждая вершина имеет в среднем равное количество соседей, а значит система уравнений (3) описывает среднестатистическую динамику количества заболеваний. Решение системы (3) можно найти в работе [5].



(a) Random network (b) Scale-free network

Рис. 4: Полностью случайный граф и безмасштабный граф (рисунок из Википедии [6]).

3.3 Модель SI для безмасштабной сети

Рассмотрим новый вид графа и уравнения для него.

Безмасштабная сеть — граф, в котором степени вершин распределены по степенному закону:

$$P(k) \sim k^{-\gamma},$$

где степень вершины — количество непосредственных соседей, $P(k)$ — доля вершин степени k, $2 < \gamma < 3$ — параметр. Примеры данного вида графов из жизни — граф авиаперелетов, граф финансовых операций, граф соавторов статей.

Распишем уравнения отдельно для вершин степени k:

$$\begin{aligned} \frac{dn_s^k(t)}{dt} &= -\beta k n_s^k(t)\Theta(\beta); \\ \frac{dn_i^k(t)}{dt} &= \beta k(N^k - n_i^k(t))\Theta(\beta); \end{aligned} \tag{4}$$

где в первом уравнении (для второго уравнения величины определены соответственно)

- $n_s^k = N^k - n_i^k(t)$ — количество здоровых среди вершин степени k,
- $\Theta(\beta)$ — вероятность того, что сосед инфицирован,
- $\beta k\Theta(\beta)$ — вероятность заражения для вершины степени k.

Более подробно, как расписываются для безмасштабных графов $\Theta(\beta)$ и решения системы можно посмотреть в работе [7].

3.4 Симуляция эпидемии на произвольном графе

Граф «Мир тесен» и безмасштабный граф — наиболее часто встречающиеся в литературе по распространению эпидемий виды графов. Это объясняется тем, что в работах [5, 7] было выяснено, что поведение эпидемий сильно для них различаются. При этом любой случайных граф может обладать свойствами как графа «Мир тесен», так и безмасштабного.

Для конкретной конфигурации социальных связей пользователей наиболее просто использование стохастических хождений по графу в качестве симуляции распространения эпидемии.

Пусть есть социальный граф, каждая вершина которого имееет одну из меток: s – человек здоров; i – человек болен; β — вероятность заражения здорового при контакте с больным. Тогда шаг алгоритма распространения болезни реализуется таким образом:

1. Выбираем вершину с меткой i и генерируем случайное число $x \in (0, 1)$.
2. Если $x < \beta$, то выбираем случайную соседнюю соседнюю вершину. Если ее метка – s, то меняем ее на i.

4 Стратегии многошагового прогнозирования временных рядов

Рассмотрим различные стратегии многошагового прогнозирования временных рядов. Далее в секции 5 задача предсказания множества больных на графе будет сведена к совокупности задач прогнозирования временного ряда.

Задача многошагового прогнозирования временного ряда (долговременного прогнозирования) состоит в получении прогнозов на H шагов вперед $[y_{T+1}, y_{T+2}, \ldots, y_{T+H}]$ временного ряда $[y_1, \ldots, y_T]$, состоящего из T наблюдений, где $H > 1$ определяет горизонт прогнозирования (количество шагов вперед для предсказания). Ниже воспользуемся стандартным обозначением из [8,9], где f и F — функции, моделирующие зависимость между прошедшими и будущими наблюдениями, d соответствует количеству наблюдений прошлого, на которые опирается предсказание, а w представляет собой слагаемое, включающее ошибку модели и внешние возмущения/шум.

4.1 Итерационный подход

Наиболее старая и интуитивная стратегия — рекурсивная (или итерационная, [10]). В данном подходе обучается одна модель f для выполнения предсказания на один шаг вперед:

$$y_{t+1} = f(y_t, \ldots, y_{t-d+1}) + w, \tag{5}$$

где $t \in d, \ldots, T-1$.

Для получения с её помощью долговременного прогноза: первый прогноз получается простым применением модели и подставляется снова в ту же модель для получения следующего прогноза (и так далее). Пусть обученная модель $\hat{f}$, тогда формально прогнозы получаются:

$$\hat{y}_{T+h} = \begin{cases} \hat{f}(y_T, \ldots, y_{T-d+1}), & \text{если } h = 1; \\ \hat{f}(\hat{y}_{T+h-1}, \ldots, \hat{y}_{T+1}, y_T, \ldots, y_{T-d+h}), & \text{если } h \in \{2, \ldots, d\}; \\ \hat{f}(\hat{y}_{T+h-1}, \ldots, \hat{y}_{T+h-d}), & \text{если } h \in \{d+1, \ldots, H\}. \end{cases} \tag{6}$$

Её основными особенностями являются:

- при итерационном подходе предсказание с предыдущей итерации подается на вход предиктору на текущей итерации;
- признаки на входе состоят как из настоящих наблюдений, так и из предсказаний с предыдущих итераций;
- итеративная процедура приводит к аккумуляции ошибки за счет ошибки прдсказаний на предыдущих шагах;
- модель склонна к низкому качеству в задаче долгосрочного предсказания, так как она оптимизируется только на одношаговых предсказаниях и не способна учитывать изменения в тренде.

Но несмотря на недостатки, рекурсивная стратегия успешно применяется для многих реальных временных рядов разными моделями машинного обучения (как, например, рекурсивными нейронными сетями в [11]).

4.2 Прямой подход

Прямой подход (впервые предложен в [10]) состоит в прогнозировании каждого горизонта независимо от других. Другими словами, H моделей f_h обучаются каждая для своего горизонта по временному ряду $[y_1, \ldots, y_T]$:

$$y_{t+h} = f_h(y_t, \ldots, y_{t-d+1}) + w, \tag{7}$$

где $t \in \{d, \ldots, T-H\}$ и $h \in \{1, \ldots, H\}$.

Предсказания получают применением H обученных моделей $\hat{f}_h$:

$$\hat{y}_{T+h} = \hat{f}_h(y_T, \ldots, y_{T-d+1}). \tag{8}$$

Таким образом:

- прямая стратегия не использует аппроксимируемые величины для получения прогнозов и не подвержена аккумуляции ошибок;
- H моделей обучаются независимо, а значит не используется зависимость соседних предсказаний, что негативно влияет на точность.

В работе [12] прямой подход успешно реализован для алгоритма нейронных сетей в задаче многошагового прогнозирования.

4.3 DirRec

DirRec стратегия (предложена в [13]), как следует из названия, комбинирует архитектуры и принципы прямого и итеративного подходов ("direct"(Dir) — "прямой"в переводе с английского, "recursive"(Rec) — "рекурсивный"). DirRec предполагает обучение H моделей, каждая для прогнозирования своего горизонта, и на каждом шаге модель дополняет множество входных наблюдений прогнозом с предыдущего шага. Таким образом длина учитываемой истории значений d модели различается для разных горизонтов. Другими словами, DirRec стратегия обучает H моделей f_h на основе наблюдений $[y_1, \ldots, y_T]$ ряда:

$$y_{t+h} = f_h(y_{t+h-1}, \ldots, y_{t-d+1}) + w, \tag{9}$$

где $t \in \{d, \ldots, T-H\}$ и $h \in \{1, \ldots, H\}$.

На этапе предсказания H обученных моделей используются следующим образом:

$$\hat{y}_{T+h} = \begin{cases} \hat{f}_h(y_T, \ldots, y_{T-d+1}), & \text{если } h = 1; \\ \hat{f}_h(\hat{y}_{T+h-1}, \ldots, \hat{y}_{T+1}, y_T, \ldots, y_{T-d+1}) & \text{если } h \in \{2, \ldots, H\}. \end{cases} \tag{10}$$

Данная стратегия превзошла оба предыдущих подхода на двух реальных временных рядах энергопотребления Санта Фе и Польши, что описано в [13].

4.4 MIMO

Три предыдущие стратегии опирались на модели с одним выходом. MIMO (предложен в [14]) расшифровывается как *Multi–Input Multi–Output* стратегия, так как используемая модель имеет как несколько входных переменных, так и несколько выходных. Такой подход мотивируется тем, что уход от моделей с одним выходом пренебрегает стохастической зависимостью между наблюдениями временного ряда в будущем, что влияет на точность прогнозирования.

Стратегия MIMO предполагает обучение одной многовыходной модели F на истории наблюдений $[y_1, \ldots, y_T]$:

$$[y_{t+H}, \ldots, y_{t+1}] = F(y_t, \ldots, y_{t-d+1}) + \mathbf{w}, \tag{11}$$

где $t \in \{d, \ldots, T-H\}$, $F : \mathrm{R}^d \to \mathrm{R}^H$ является векторной функций, $\mathbf{w} \in \mathrm{R}^{\mathbf{H}}$ — вектор шума с не обязательно диагональной ковариационной матрицей.

Предсказания получаются одним действием:

$$[\hat{y}_{T+H}, \ldots, \hat{y}_{T+1}] = \hat{F}(y_T, \ldots, y_{T-d+1}). \tag{12}$$

Стратегия MIMO избегает предположения об условной независимости наблюдений, лежащего в основе прямого подхода, а также аккумуляции ошибки при рекурсивном подходе. Но прогнозирование всех горизонтов с помощью одной модели снижает гибкость модели. Поэтому была введена новая многовыходная модель DIRMO.

4.5 DIRMO

DIRMO создавалась, чтобы иметь свойства как прямой, так и MIMO стратегии (статья [15]). DIRMO прогнозирует горизонт H по блокам, где каждый блок предсказывается в рамках MIMO стратегии. Таким образом, H-шаговое прогнозирование разбивается на $n = \frac{H}{s}$ задач прогнозированя, каждая содержит модель F_p с количеством выходных переменных, равным $s \in \{1, \ldots, H\}$, s — гиперпараметр.

$$[y_{t+p*s}, \ldots, y_{t+(p-1)*s+1}] = F_p(y_t, \ldots, y_{t-d+1}) + \mathbf{w}, \tag{13}$$

где $t \in \{d, \ldots, T-H\}$, $p \in \{1, \ldots, n\}$ и $F_p : \mathrm{R}^d \to \mathrm{R}^s$ является векторной функцией в случае $s > 1$.

	$\hat{y}_{T+1}$	$\hat{y}_{T+2}$	$\hat{y}_{T+3}$	$\hat{y}_{T+4}$
Итерационный подход	$\hat{f}_h(y_T, \ldots, y_{T-d+1})$	$\hat{f}_h(\hat{y}_{T+1}, y_T, \ldots, y_{T-d+2})$	$\hat{f}_h(\hat{y}_{T+2}, \hat{y}_{T+1}, \ldots, y_{T-d+3})$	$\hat{f}_h(\hat{y}_{T+3}, \hat{y}_{T+2}, \ldots, y_{T-d+4})$
Прямой подход	$\hat{f}_1(y_T, \ldots y_{T-d+1})$	$\hat{f}_2(y_T, \ldots, y_{T-d+1})$	$\hat{f}_3(y_T, \ldots, y_{T-d+1})$	$\hat{f}_4(y_T, \ldots, y_{T-d+1})$
DirRec	$\hat{f}_1(y_T, \ldots, y_{T-d+1})$	$\hat{f}_2(\hat{y}_{T+1}, y_T, \ldots, y_{T-d+1})$	$\hat{f}_3(\hat{y}_{T+2}, \hat{y}_{T+1}, \ldots, y_{T-d+1})$	$\hat{f}_4(\hat{y}_{T+3}, \hat{y}_{T+2}, \ldots, y_{T-d+1})$
MIMO	$\hat{F}(y_T, \ldots, y_{T-d+1})$			
DIRMO ($s = 2$)	$\hat{F}_1(y_T, \ldots, y_{T-d+1})$		$\hat{F}_2(y_T, \ldots, y_{T-d+1})$	

Таблица 1: Модели, используемые в различных стратегиях для прогнозирования на 4 шага вперед, [8].

	Количество моделей	Количество выходных переменных	Вычислительное время
Итерационный подход	1	1	$1 \times T_f$
Прямой подход	H	1	$H \times T_f$
DirRec	H	1	$H \times (T_f + const)$
MIMO	1	H	$1 \times T_F$
DIRMO	$\frac{H}{s}$	s	$\frac{H}{s} \times T_F$

Таблица 2: Для каждой стратегии прогнозирования: количество и тип обучаемых моделей (с одним или несколькими выходными переменными), оценка вычислительного времени, [8].

Действительно, для случая $s = 1$ количество разбиваемых блоков n равно H, что является в чистом виде прямым подходом из раздела 4.2.

Прогнозы с горизонтами $1, \ldots, H$ возвращаются n обученными моделями $\hat{F}_p$:

$$[\hat{y}_{T+p*s,\ldots,y_{T+(p-1)*s+1}}] = \hat{F}_p(y_T, \ldots y_{T-d+1}). \tag{14}$$

DIRMO была успешно применена в конкурсах прогнозирования ESTSP'07 [15] и NN3 [16].

4.6 Сравнение стратегий

Были представлены пять возможных стратегий долговременного прогнозирования временных рядов: итерационный, прямой, DirRec, MIMO и DIRMO. Как было сказано, DirRec–стратегия является комбинацией прямого и итерационного подходов, а DIRMO — комбинацией прямого и MIMO-подхода.

В зависимости от выбора стратегии меняет количество, а также тип обучаемых моделей.

В качестве примера разберем задачу прогнозирования временного ряда $[y_1, \ldots, y_T]$, где горизонт прогнозирования H равен 4. Таблица 1 для каждой стратегии отображает входные значения и используемые модели для вычисления четырех прогнозов $[\hat{y}_{T+1}, \hat{y}_{T+2}, \hat{y}_{T+3}, \hat{y}_{T+4}]$.

Оценим теперь вычислительное время построения моделей для каждой стратегии. Пусть T_f и T_F — количество единиц времени, необходимое для обучения модели с одним выходом и с несколькими выходами соответственно. Для задачи прогнозирования с горизонтом H составлена таблица 2, где отображены количество, тип (с одним выходом или несколькими) обучаемых моделей и требуемое вычислительное время.

Предположим, что $T_F = T_f + \delta$, что логично, так как обучение модели с несколькими выходами должно обучаться дольше аналогичной модели с одним выходом. Тогда можем отранжировать стратегии по возрастанию затрачиваемого вычислительного времени:

$$\underbrace{1 \times T_f}_{\text{Итерационнный}} < \underbrace{1 \times T_F}_{\text{MIMO}} < \underbrace{\frac{H}{s} \times T_F}_{\text{DIRMO}} < \underbrace{H \times T_f}_{\text{Прямой}} < \underbrace{H \times (T_f + \mu)}_{\text{DirRec}}, \tag{15}$$

где предполагается, что параметр s DIRMO–стратегии не равен 1 или H.

Стоит заметить, что DIRMO требует тюнинга параметра s, что превращает её в самую вычислительно ёмкую по времени. Также время обучения модели с одним выходом DirRec стратегии равно $T_f + \mu$, так как количество входных переменных увеличивается на каждом шаге.

5 Предлагаемый подход

Модель SI тяжело расширить для случаев, когда нужно учесть дополнительные факторы, влияющие на распространение болезней. Как то: количество болеющих, с которым взаимодействует здоровый, или географическое местоположение, или сообщества. Для учета такого рода факторов в данной работе предлагается использовать методы машинного обучения.

5.1 Основные обозначения

Определим в момент времени t для i–ого пользователя

- $y_t^i \in \{0, 1\}$ — его метку (1 — здоров, 0 — болен);
- $\mathbf{a}_t^i \in \mathrm{R}^{1 \times D}$ — столбец признаков пользователя,

где D – количество признаков, описывающих человека и структуру текущего графа (далее извлекаемые признаки будут описаны подробнее). Отметим, что в признаковое описание пользователя может входить как его профиль, так и характеристики состояния графа (например, метки его соседей).

Назовем состоянием социального графа $G = (V, E)$ метки его вершин. Каждому состоянию в момент времени t соответствует

- столбец меток $\mathbf{y}_t = \{y_t^i\}_{i=1,\ldots,N} \in \{0, 1\}^{1 \times N}$ (0 – человек здоров, 1 – болен)
- и матрица состояния $\mathbf{A}_t = \{(\mathbf{a}_t^i)^T\}_{i=1,\ldots,N} \in R^{N \times D}$,

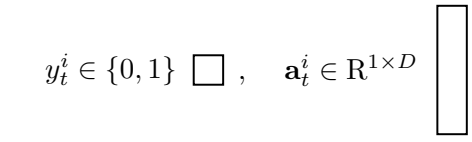


Рис. 5: Визуализация метки и столбца признаков y_t^i, $\mathbf{a}_t^i$ одного пользователя.

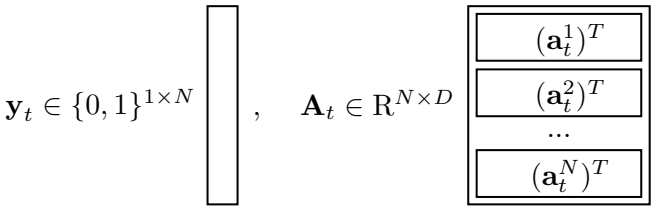


Рис. 6: Визуализация меток вершин графа $\mathbf{y_t}$, матрицы состояния графа $\mathbf{A_t}$.

где N – количество человек (вершин).

Известна история заболеваний за T моментов времени:

$$\text{пары } (\mathbf{A}_t, \mathbf{y}_t),$$

где $t \in \{1, \dots, T\}$.

Задача состоит в прогнозировании меток вершин на H шагов вперед:

$$\mathbf{y}_{T+t},$$

где $t \in \{1, \dots, H\}$.

По аналогии с статегиями прогнозирования временных рядов из раздела 4 построим модели машинного обучения для предсказания метки текущей вершины. Будем рассматривать в качестве объекта выборки: одного пользователя в графе в t-ом состоянии.

5.2 Итерационный подход

Как описано в разделе 4.1 про итерационный подход, пусть построена модель машинного обучения для предсказания на один шаг вперед метки пользователя:

$$\hat{y}_{t+1} = \hat{f}(\mathbf{a}_t, y_t, \dots, \mathbf{a}_{t-d+1}, y_{t-d+1}), \tag{16}$$

где $t \in \{d, \dots, T-1\}$, $\mathbf{a}_t \in \mathrm{R}^{1\times D}$ – признаки пользователя в момент времени t, $d < T$ — параметр модели.

Прогноз метки на h шагов вперед выполняется следующим образом:

$$\hat{y}_{T+h} = \begin{cases} \hat{f}(\mathbf{a}_T, y_T, \dots, \mathbf{a}_{T-d+1}, y_{T-d+1}), & \text{если } h = 1; \\ \hat{f}(\hat{\mathbf{a}}_{T+h-1}, \hat{y}_{T+h-1}, \dots, \hat{\mathbf{a}}_{T+1}, \hat{y}_{T+1}, \mathbf{a}_T, y_T, \dots, \mathbf{a}_{T-d+h}, y_{T-d+h}), & \text{если } h \in \{2, \dots, d\}; \\ \hat{f}(\hat{\mathbf{a}}_{T+h-1}, \hat{y}_{T+h-1}, \dots, \hat{\mathbf{a}}_{T+h-d}, \hat{y}_{T+h-d}), & \text{если } h \in \{d+1, \dots, H\}. \end{cases} \tag{17}$$

«Крышка» в обозначении $\hat{\mathbf{a}}_{t+h}$ подчеркивает, что признаки пользователя в $(t+h)$-ый момент времени используют прогнозы меток вершин графа $[\hat{y}_{t+h}^1, \dots, \hat{y}_{t+h}^N]$. Таким образом, прогноз для i-ого пользователя зависит от меток всех пользователей в предыдущие моменты времени. Это будет верно и для остальных стратегий.

Процесс обучения построим на основе процесса построения предсказаний. Обучающая выборка будет состоять из $T-d$ блоков (признаковая матрица блока, верные метки):

$$\big([\mathbf{A}_{d+k-1}, \mathbf{y}_{d+k-1}, \dots, \mathbf{A}_k, \mathbf{y}_k], \mathbf{y}_{d+k}\big),$$

где $k \in \{1, \dots, T-d\}$ — номер блока. Назовем признаковой матрицей k-ого блока $\mathbf{X}_k = [\mathbf{A}_{d+k-1}, \mathbf{y}_{d+k-1}, \dots, \mathbf{A}_k, \mathbf{y}_k] \in \mathrm{R}^{N\times d(D+1)}$, тогда финальная обучающая выборка для модели формируется конкатенацией матриц блока и соответсвующих меток

$$\big([\mathbf{X}_1, \dots, \mathbf{X}_{T-d}]^T, [\mathbf{y}_{d+1}, \dots, \mathbf{y}_T]^T\big).$$

На рис. 7, 8 визуализированы матрица блока $\mathbf{X}_k$ и финальная обучающая выборка $(\mathbf{X}, \mathbf{y})$ для итерационного подхода.

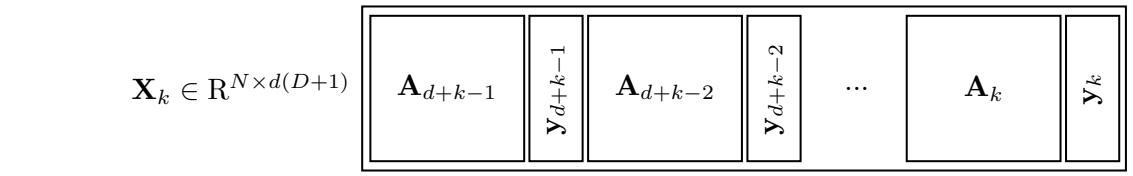


Рис. 7: Визуализация матрицы блока итерационного подхода $\mathbf{X}_k$ как конкатенации матриц состояний и столбцов меток $[\mathbf{A}_{d+k-1}, \mathbf{y}_{d+k-1}, \mathbf{A}_{d+k-2}, \mathbf{y}_{d+k-2}, \ldots, \mathbf{A}_k, \mathbf{y}_k]$.

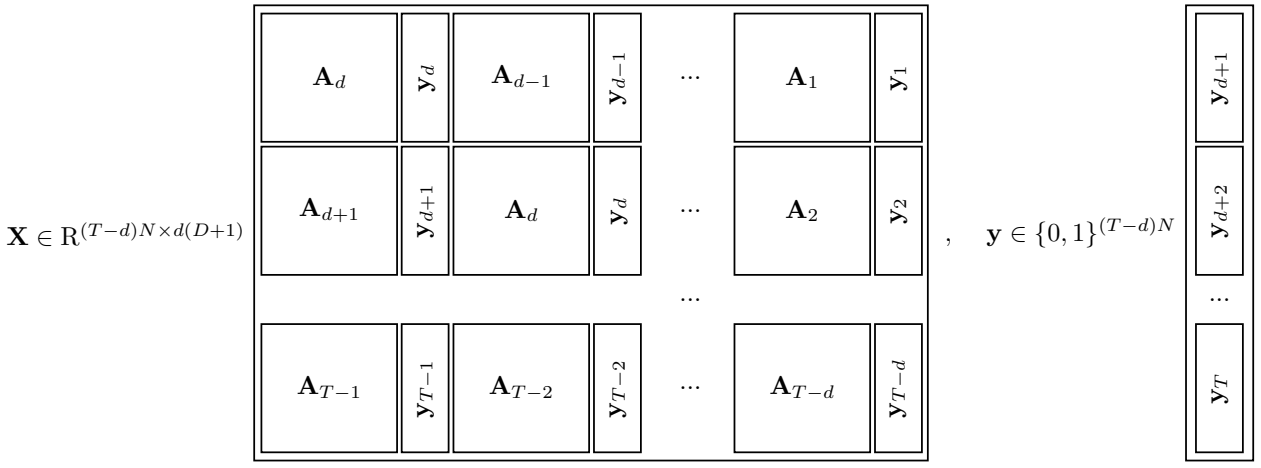


Рис. 8: Обучающая выборка $(\mathbf{X}, \mathbf{y})$ итерационного подхода.

5.3 Прямой подход

Как описано в разделе 4.2 в прямом подходе обучается H моделей $\hat{f}_h$ для каждого горизонта:

$$\hat{y}_{t+h} = \hat{f}_h(\mathbf{a}_t, y_t, \ldots, \mathbf{a}_{t-d+1}, y_{t-d+1}), \tag{18}$$

где $t \in \{d, \ldots, T-H\}$ и $h \in \{1, \ldots, H\}$, $d < T$ — параметр модели.

Прогноз метки пользователя получается применением $\hat{f}_h$:

$$\hat{y}_{T+h} = \hat{f}_h(\mathbf{a}_T, y_T, \ldots, \mathbf{a}_{T-d+1}, y_{T-d+1}). \tag{19}$$

Процесс обучения построим независимым для каждой из моделей f_h. Для h–ой модели обучающая выборка будет состоять из $T-h-d+1$ блоков (признаковая матрица блока, верные метки):

$$\big([\mathbf{A}_{T-h-k+1}, \mathbf{y}_{T-h-k+1}, \ldots, \mathbf{A}_{T-h-k-d+2}, \mathbf{y}_{T-h-k-d+2}], \mathbf{y}_{T-k+1}\big),$$

где $k \in \{1, \ldots, T-h-d+1\}$ — номер блока. Назовем признаковой матрицей k–ого блока $\mathbf{X}_k = [\mathbf{A}_{T-h-k+1}, \mathbf{y}_{T-h-k+1}, \ldots, \mathbf{A}_{T-h-k-d+2}, \mathbf{y}_{T-h-k-d+2}]$, тогда обучающая выборка h-ой модели формируется конкатенацией матриц блока и соответствующих меток:

$$\big([\mathbf{X_1}, \ldots, \mathbf{X}_{T-h-d+1}]^T, [\mathbf{y}_T, \ldots, \mathbf{y}_{d+h}]^T\big).$$

5.4 DirRec

DirRec-стратегия похожа на прямую стратегию, но использует в качестве входных переменных прогнозы предыдущих моделей. Пусть обучены H моделей $\hat{f}_h$:

$$\hat{y}_{t+h} = \hat{f}_h(\mathbf{a}_{t+h-1}, y_{t+h-1}, \ldots, \mathbf{a}_{t-d+1}, y_{t-s+1}), \tag{20}$$

где $t \in \{d, \ldots, T-H\}$ и $h \in \{1, \ldots, H\}$, d — параметр модели.

Прогноз получают

$$\hat{y}_{T+h} = \begin{cases} \hat{f}_h(\mathbf{a}_T, y_T, \ldots, \mathbf{a}_{T-d+1}, y_{T-d+1}), & \text{если } h = 1; \\ \hat{f}_h(\hat{\mathbf{a}}_{T+h-1}, \hat{y}_{T+h-1}, \ldots, \hat{\mathbf{a}}_{T+1}, \hat{y}_{T+1}, \mathbf{a}_T, y_T, \ldots, \mathbf{a}_{T-d+1}, y_{T-d+1}), & \text{если } h \in \{2, \ldots, H\}. \end{cases} \tag{21}$$

Процесс обучения разобьем на независимое обучение H функций. Отметим, что он повторяет обучение прямого подхода с той разницей, что для каждого h свое значение d. Обучающая выборка для $\hat{f}_h$ делится на $T-h-d+1$ блоков (признаковая матрица блока, верные метки):

$$\big([\mathbf{A}_{T-h-k+1}, \mathbf{y}_{T-h-k+1}, \dots, \mathbf{A}_{T-h-k-d+2}, \mathbf{y}_{T-h-k-d+2}], \mathbf{y}_{T-k+1}\big),$$

где $k \in \{1, \dots, T-h-d+1\}$ — номер блока. Назовем признаковой матрицей k–ого блока $\mathbf{X}_k = [\mathbf{A}_{T-h-k+1}, \mathbf{y}_{T-h-k+1}, \dots, \mathbf{A}_{T-h-k-d+2}, \mathbf{y}_{T-h-k-d+2}]$, тогда обучающая выборка h-ой модели формируется конкатенацией матриц блока и соответствующих меток:

$$\big([\mathbf{X_1}, \dots, \mathbf{X}_{T-h-d+1}]^T, [\mathbf{y}_T, \dots, \mathbf{y}_{d+h}]^T\big).$$

5.5 MIMO

MIMO предполагает обучение одной модели $\hat{F}$ с несколькими выходными значениями

$$[\hat{y}_{t+H}, \dots, \hat{y}_{t+1}] = \hat{F}(\mathbf{a}_t, y_t, \dots, \mathbf{a}_{t-d+1}, y_{t-d+1}), \tag{22}$$

где $t \in \{d, \dots, T-H\}$, d – параметр модели.

Прогноз получают одним действием:

$$[\hat{y}_{T+H}, \dots, \hat{y}_{T+1}] = \hat{F}(y_T, \dots, y_{T-d+1}). \tag{23}$$

Процесс обучения для схемы MIMO проходит быстрее, чем для DirRec. Обучающая выборка разбивается на $T-d-H+1$ блоков. Пара (признаковая матрица, верные метки) для каждого блока выглядят:

$$\big([\mathbf{A}_{T-H-d-k+2}, \dots, \mathbf{A}_{T-H-k+1}], [\mathbf{y}_{T-H-k+2}, \dots, \mathbf{y}_{T-k+1}]\big),$$

где $k \in \{1, \dots, T-d-H+1\}$ — номер блока. Аналогично разделам выше назовем матрицей блока $\mathbf{X}_k = [\mathbf{A}_{T-H-d-k+2}, \dots, \mathbf{A}_{T-H-k+1}]$, а также $\mathbf{Y}_k = [\mathbf{y}_{T-H-k+2}, \dots, \mathbf{y}_{T-k+1}]$, тогда полная обучающая выборка представима в виде:

$$\big([\mathbf{X}_1, \dots, \mathbf{X}_{T-d-H+1}]^T, [\mathbf{Y}_1, \dots, \mathbf{Y}_{T-d-H+1}]^T\big).$$

5.6 DIRMO

В DIRMO–стратегии обучается $n = \frac{H}{s}$ моделей $\hat{F}_p$

$$[\hat{y}_{t+p*s}, \dots, \hat{y}_{t+(p-1)*s+1}] = \hat{F}_p(\mathbf{a}_t, y_t, \dots, \mathbf{a}_{t-d+1}, y_{t-d+1}), \tag{24}$$

где $t \in \{d, \dots, T-H\}$, $p \in \{1, \dots, n\}$, s и d — параметры модели.

Прогноз возвращается n моделями:

$$[\hat{y}_{T+p*s}, \dots, \hat{y}_{T+(p-1)*s+1}] = \hat{F}_p(\mathbf{a}_T, y_T, \dots, \mathbf{a}_{T-d+1}, y_{T-d+1}). \tag{25}$$

Процесс обучения построен аналогично обучению MIMO, так как каждая из n моделей строится независимо аналогично F из MIMO. Для построения $\hat{F}_p$ разобьем обучающую выборку на $T-d-p*s+1$ блоков. Пары (признаковая матрица блока, верные метки) выглядят:

$$\big([\mathbf{A}_k, \mathbf{y}_k, \dots, \mathbf{A}_{k+d-1}, \mathbf{y}_{k+d-1}], [\mathbf{y}_{k+(p-1)*s+1}, \dots, \mathbf{y}_{k+p*s}]\big),$$

где $k \in \{1, \dots, T-d-p*s+1\}$ — номер блока. Тогда обучающая выборка для $\hat{F}_p$ получается конкатенацией признаковых матриц блоков и соответвующих меток. Назовем $\mathbf{X}_k = [\mathbf{A}_k, \mathbf{y}_k, \dots, \mathbf{A}_{k+d-1}, \mathbf{y}_{k+d-1}]$ и $\mathbf{Y}_k = [\mathbf{y}_{k+(p-1)*s+1}, \dots, \mathbf{y}_{k+p*s}]$, тогда обучающая выборка:

$$\big([\mathbf{X}_1, \dots, \mathbf{X}_{T-d-p*s+1}]^T, [\mathbf{Y}_1, \dots, \mathbf{Y}_{T-d-p*s+1}]^T\big).$$

6 Анализ распространения Twitter-постов с тегом «Boson Higgs»

6.1 Описание и предобработка данных

В датасете SNAP:Higgs Twitter Dataset известны:

- история Twitter-сообщений с тэгом «Higgs Boson» за 7 дней;
- социальный граф связей пользователей;
- профили пользователей.

Объемы данных:

$\sim$ 450000 пользователей;

$\sim$ 300000 пользователей, которые хоть раз постили сообщение о бозоне, остальные — их друзья;

$\sim$ 5600000 сообщений.

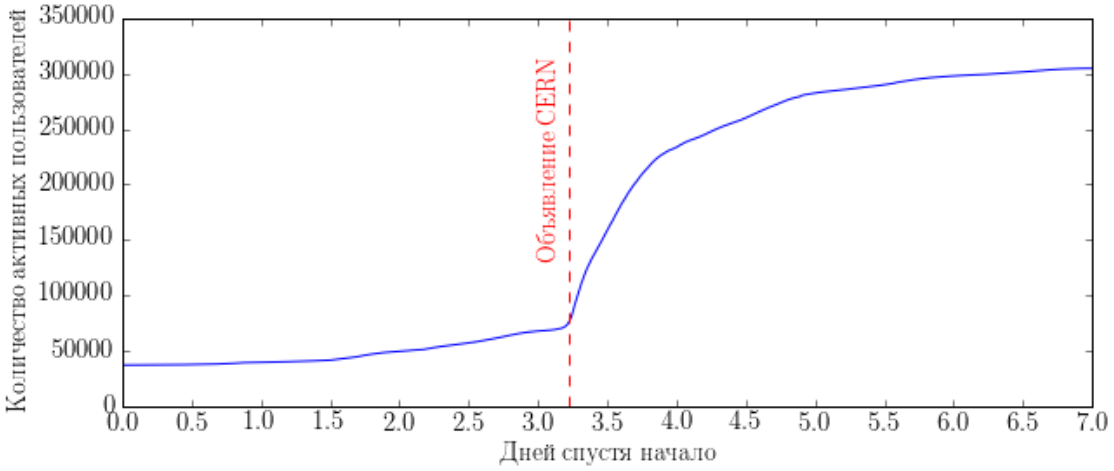


Рис. 9: Количество пользователей, запостивших новость о Бозоне хоть раз за предшествующие моменты времени (априори неубывающая функция).

Рассмотрим задачу прогнозирования меток вершин твиттер-графа в будущее на основе дневной истории наблюдений. Значению $y_t = 1$ метки вершины («болен») в t–ый момент времени соответсвует то, что **пользователь хоть раз за t-1 моментов времени опубликовал пост с тегом «Boson Higgs»**, значению $y_t = 0$ («здоров») соответсвует то, что **пользователь до текущего момента не публиковал новость о бозоне.**

На рисунке 9 отображена динамика количества людей, запостивших хоть раз за предшествующие моменты времени (априори неубывающая функция) новость о Бозоне Хиггса. На третий день в 8 часов учеными из CERN было сделано объявление о существовании новой частицы, что повлекло большое количество постов на эту тему.

Чтобы выделить дискретные состояния графа из данных, разьбьем 3.6 дня на равные периоды по 4 часа и объявим t-ым состоянием графа его метки к концу t-ого периода. Тогда образуется 24 состояний, на первых 5 из которых будет проходить обучение и тюнинг параметров, и для оставшихся 19 будем делать прогноз.

Для чистоты эксперимента исключим внешние факторы из данных: будем рассматривать динамику распространения твиттер-постов только после объявления CERN.

6.2 Признаки

В A_t матрицу признаков в момент времени t вершины графа входит зависимое от данных описание человека, номер шага в модели рекурсивного получения предсказаний, а также следующие характеристики состояния графа:

- количество подписчиков;
- количество пользователей, на которых подписан данный человек;
- отношения количества подписчиков к количеству пользователй, на которых подписан (мера общительности);
- количество «больных» непосредственных знакомых;
- количество «больных» людей, длина пути в графе до которых не превышает 2;
- доля «больных» среди непосредственных знакомых;
- доля «больных» среди людей, длина пути в графе до которых не превышает 2;
- длина пути в графе до ближайшего «больного»;
- количество единиц времени, в течение которого ближайший «больной» является больным;
- доля «больных» среди знакомых ближайшего «больного»;
- прирост «больных» среди непосредственных знакомых за последний период;
- плотность подграфа, состоящего из всех знакомых знакомых.

6.3 Сравнение алгоритмов

Для выбора лучшего алгоритма машинного обучения разобьем обучающую выборку на обучающую и валидационную, и применим к ним одношаговый подход, опирающийся на два последних наблюдения. Для простоты тестирование проводим с помощью итерационного подхода, в секции 6.4 же будут сравниваться разные подходы прогнозирования для лучшего алгоритма.

В таблице 3 приведены значения метрик качества на валидационной выборке для разных алгоритмов, жирным выделены лучшие значения.

Параметры моделей валидировались по подвыборке 100000 случайных вершин обучающей выборки, качество финальной модели оценивалось по оставшимся вершинам. Алгоритмы взяты из библиотеки `sklearn` языка Python. Параметры приведенных в таблице 3 моделей:

- `LogisticRegression(penalty="l2", C=1, max_iter=100)` – логистическая регрессия;
- `Perceptron(alpha=0.0, n_iter=5, eta0=1.0)` — персептрон;
- `KNeighborsClassifier(n_neighbors=5, weights="uniform", p=2, metric="minkowski")` — алгоритм ближайших соседей;
- `GaussianNB()` — наивный байес;
- `RandomForestClassifier(n_estimators=50, criterion="gini", max_depth=60, class_weight="subsamp` — случайный лес.

Логистическая регрессия превосходит по всем метрикам, далее воспользуемся ею.

	Логистическая регрессия	Персептрон	kNN	Наивный Байес	Random Forest
accuracy	**0.95**	0.6	0.6	0.58	0.88
recall	**1**	0.88	0.48	0.16	0.83
precision	**0.92**	0.55	0.61	0.86	**0.92**
f-score	**0.95**	0.68	0.53	0.27	0.87
AUC	**0.96**	0.61	0.59	0.57	0.88

Таблица 3: Качество работы алгоритмов.

6.4 Сравнение стратегий прогнозирования

Для тестирования возьмем подмножество из 100000 пользователей обучающей выборки и сравним на ней стратегии прогнозирования.

На обучающей выборке из 5 состояний можно обучить модель из итерационного подхода для $d = 1$, $d = 2$, $d = 3$, $d = 4$. Лучшее качество на валидационной выборке (таблица 4) оказалось для модели, опирающейся на два последних наблюдения. При $d = 4$ качество упало значительно — это можно объяснить, например, тем, что обучающая выборка ограничена длиной известной истории, и количество обучающих примеров для $d = 4$ втрое меньше, чем для $d = 2$, и данных не хватило, чтобы настроиться на историю такой длины.

Прямой подход показал лучшую точность предсказания больных по сравнению с итерационным, но проигрывает по точности всех ответов алгоритма. DirRec сходен по качеству с итерационным подходом. MIMO и DIRMO подходы основываются на идее, что модель может настраиваться на выход с несколькими переменными, логистическая регрессия таким свойством не обладает, поэтому не будем включать эти стратегии в сравнение.

	Итерационный подход				Прямой подход	DirRec
	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 3$, $H = 2$	$d_1 = 3$, $d_2 = 4$, $H = 2$
accuracy	0.63	**0.97**	0.96	0.83	0.83	0.96
recall	0.91	**1**	**1**	0.75	0.73	0.97
precision	0.65	0.96	0.94	0.93	**0.98**	0.96
f-score	0.76	**0.98**	0.97	0.83	0.83	0.94
AUC	0.52	**0.97**	0.96	0.84	0.44	0.95

Таблица 4: Качество на валидационной выборки логистической регрессии для разных стратегий.

Подсчитано время обучения (усредненное по 20 запускам) для разных моделей (рис. 10). В целом, подходы близки по скорости к теоретическим оценкам(15): итерационный подход, в среднем, самый быстрый, прямой подход медленнее, DirRec наиболее вычислительно затратный.

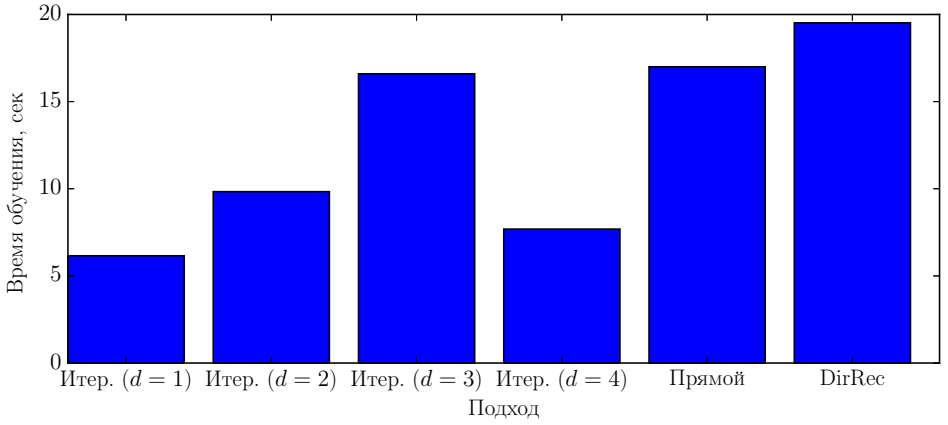


Рис. 10: Время обучения логистической регрессии для разных стратегий.

6.5 Сравнение моделей

На рисунке 12 отображено количество «больных» пользователей в зависимости от времени для трех моделей, в таблице 5 — качество моделей.

Логистическая регрессия, построенная на всех признаках, показывает лучшее качество и демонстрирует схожую с истинной динамику. 100% точность предсказания «больных» означает, что модель ни разу не объявила пользователя «больным» раньше времени, 96.5% полнота показывает, что в 3.5% случаях момент «заражения» был определен с опозданием.

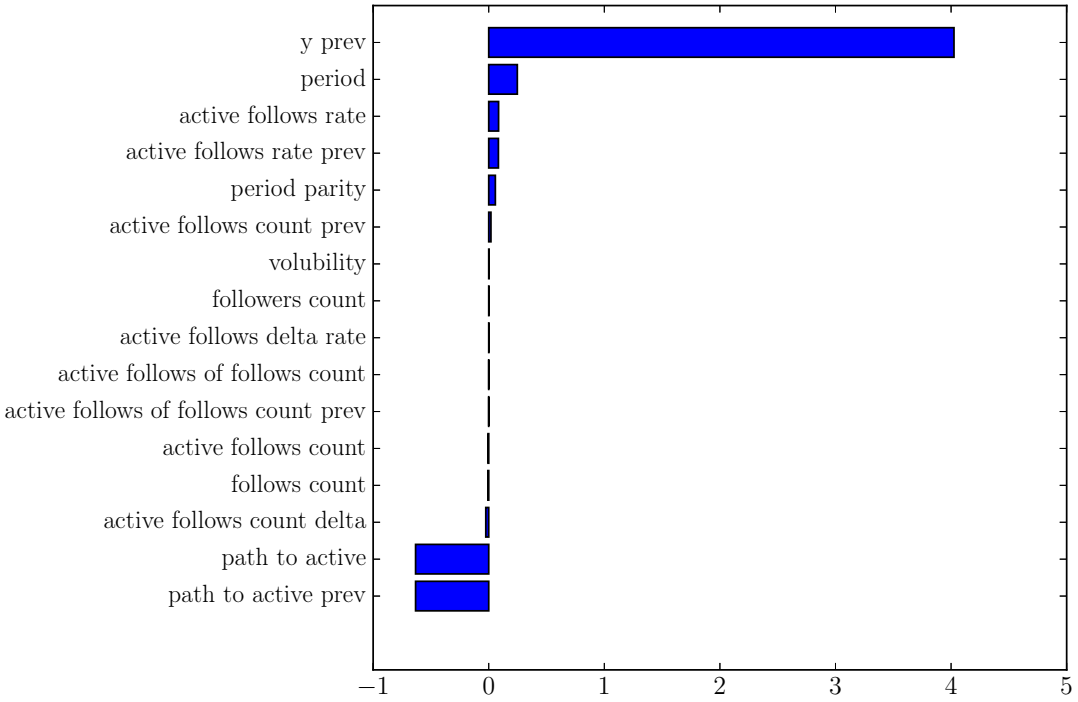


Рис. 11: Веса признаков в модели логистической регрессии.

	SI-модель	Модель машинного обучения (без профиля пользователя)	Модель машинного обучения (все признаки)
Recall	57.1%	93.2%	96.5%
Precision	67.3%	99.1%	100%
AUC	0.59	0.94	0.98

Таблица 5: Качество моделей на всей выборке (3.5 дня).

7 Результаты

В ходе выполнения работы были рассмотрены несколько стратегий прогнозирования временных рядов для модели машинного обучения в рамках задачи предсказания распространения новости о Бозоне Хиггса Твиттер–сети, лучшее качество (как например, условие «нездоровления» пользователей) показала рекурсивная одношаговая модель, опирающаяся на историю двух последних наблюдений.

Логистическая регрессия в данной задаче превзошла по метрике AUC модели случайного леса, персептрона и наивного байеса засчет бóльшей полноты.

Из рассмотренных эпидемиологических моделей к данному датасету применима только стохастическая SI-модель, так как социальный граф Твиттера не является примером графа «Маленький мир» или Безмасштабной сети. SI–модель проигрывает предлагаемому методу в качестве предсказания меток пользователей. Также график динамики количества «больных» SI–модели визуально менее близок к графику истинной динамики.

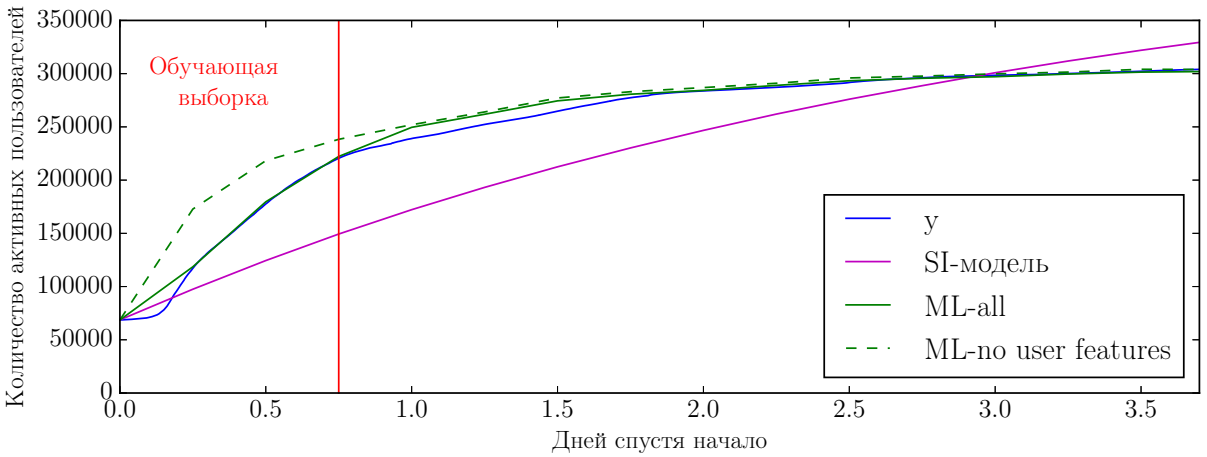


Рис. 12: Количество «больных» пользователей в зависимости от времени: истинное, предсказание SI–модели, предсказание лучшей модели машинного обучения, обученной на всех признаках и модели, обученной на всех признаках, кроме профиля пользователя.

Как видно на рисунке 11

- значение метки пользователя «1» в предыдущий момент времени увеличивает вероятность предсказать «1»,
- время дня — значимый признак, и чем ближе к вечеру, тем больше вероятность поста пользователя,
- чем длиннее путь по социальному графу до ближайшего блоггера, тем меньше вероятность поста пользователя.

Такие результаты интуитивно понятны: если метка пользователя $y_{t-1} = 1$ («болен»), то метка в следующий момент времени y_t обязана быть равной 1 (о чем модель должна догадаться по обучающей выборке).

8 Заключение

На защиту выносятся

- методы обучения моделей машинного обучения для итерационной, прямой, DirRec, MIMO и DIRMO стратегий прогнозирования в задаче предсказания меток вершин графа на H шагов вперед (и реализация на Python),
- сравнение стратегий прогнозирования по качеству и вычислительной сложности в задаче предсказания распространения новости «Boson Higgs» в графе Twitter-a,
- сравнение алгоритмов машинного обучения по качеству в задаче предсказания распространения новости «Boson Higgs» в графе Twitter-a,
- реализация SI–модели на графе (Python),
- анализ признаков, влияющих на распространение новости в графе.

Список литературы

[1] M. D. Domenico, A. Lima, P. Mougel, and M. Musolesi, "The anatomy of a scientific rumor," *Scientific Reports*, vol. 3, no. 2980, 2013.

[2] R. M. Anderson and R. M. May, "Infectious diseases of humans, dynamics and control," Texas, 1994.

[3] L. J. S. Allen, *Some Discrete-Time SI, SIR, and SIS Epidemic Models.* Oxford: Oxford Univ. Press, 1991.

[4] D. J. Watts and S. H. Strogatz, "Collective dynamics of small-world networks," *Nature*, vol. 393, pp. 440–442, 1998.

[5] C. Moore and M. E. Newman, "Epidemics and percolation in small–world networks," *Phys. Rev.*, vol. 61, pp. 5678–5682, 2000.

[6] Wikipedia, "Scale-free-network." `https://en.wikipedia.org/wiki/Scale-free_network`, 2016.

[7] R. Pastor-Satorras and A. Vespignani, "Epidemic spreading in scale-free networks," *Phys. Rev. Lett.*, vol. 86, pp. 3200–3203, 2001.

[8] S. B. Taieb, G. Bontempi, A. Atiya, and A. Sorjamaa, "A review and comparison of strategies for multi-step ahead time series forecasting based on the nn5 forecasting competition," *pre-print*, 2011. arXiv:1108.3259.

[9] G. Bontempi, "Machine learning strategies for time series prediction." `http://www.ulb.ac.be/di/map/gbonte/ftp/time_ser.pdf`, 2013.

[10] N. Gershenfeld and A. Weigend, "The future of time series: Learning and understanding. time series prediction: Forecasting the future and understanding the past," 1994.

[11] E. W. Saad, D. V. Prokhorov, and D. C. Wunsch, "Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks," *IEEE Transactions on Neural Networks*, vol. 9, pp. 1456–1470, Nov 1998.

[12] D. M. Kline and G. Zhang, "Methods for multi-step time series forecasting with neural networks," *Neural networks in business forecasting*, pp. 226–250, 2004.

[13] A. Sorjamaa and A. Lendasse, "Time series prediction using dirrec strategy," in *ESANN06, European Symposium on Artificial Neural Networks* (M. Verleysen, ed.), (Bruges, Belgium), pp. 143–148, European Symposium on Artificial Neural Networks, April 26-28 2006.

[14] G. Bontempi, "Long term time series prediction with multi-input multi-output local learning," *Proc. 2nd ESTSP*, pp. 145–154, 2008.

[15] S. B. Taieb, G. Bontempi, A. Sorjamaa, and A. Lendasse, "Long-term prediction of time series by combining direct and MIMO strategies," in *International Joint Conference on Neural Networks, IJCNN 2009, Atlanta, Georgia, USA, 14-19 June 2009*, pp. 3054–3061, 2009.

[16] S. Ben Taieb, A. Sorjamaa, and G. Bontempi, "Multiple-output modeling for multi-step-ahead time series forecasting," *Neurocomput.*, vol. 73, pp. 1950–1957, June 2010.