

# Предсказание вероятности победы Radiant в Dota2

Индивидуальный этап

Липкина Анна  
317 группа

25 ноября 2016 г.

# Входные данные

- Данные по матчам в формате csv
- Скрипт, позволяющий вытащить больше информации из json-файла матчей
- Возможность скачивать *любые* дополнительные данные

## Что юзала

- + Данные csv-файлы
- + Дополнительные данные

## А что нет

- Заленилась разбираться в скрипте

# Обработка данных

- В csv-выборке много пропущенных значений в данных. В основном, это времена каких-то событий и номера игроков.

## Как обрабатывала

- + Заменяла всё на -1

## Что еще пробовала

- Заменить всё на среднее значение по фиче. Точность становилась хуже, чем при вышеописанном способе.

# Обработка данных

- Выкинуть `lobby_type`, так как важно не то, каким способом герои были выбраны, а какие герои были выбраны
- Герои (номера героев) — категориальный признак  $\Rightarrow$  делаем `bag-of-words` по героям: -1, если герой играет за темную сторону, 1 — если за светлую.
- Сбор статистики по командным, а не индивидуальным характеристикам

# Статистика командных показателей

- Для каждого командного показателя  $team\_factor_{\{r, d\}}$  (для светлых и темных соответственно) будем добавлять в признаки следующие три величины:

$$+ team\_factor\_r - team\_factor\_d$$

$$+ \frac{team\_factor\_r}{team\_factor\_d + EPS}$$

$$+ \frac{team\_factor\_r - team\_factor\_d}{\sqrt{|team\_factor\_r| + |team\_factor\_d| + EPS}}$$

*It's a kind of magic!*

Где  $EPS = 10^{-6}$  — используется для предотвращения деления на 0.

# Командные показатели

- Введём понятие **прокаченности героя** — линейная комбинация данных в csv признаков героя. Коэффициенты подбирались вручную из «здравого смысла». Сумма коэффициентов равна 1:

$$\text{hero\_coolness} = 0.25 \cdot \text{level} + 0.15 \cdot \text{xp} + 0.1 \cdot \text{gold} + 0.05 \cdot \text{lh} + 0.3 \cdot \text{kills} - 0.3 \cdot \text{deaths} + 0.05 \cdot \text{items}$$

Для нахождения командного показателя прокаченности просто просуммируем прокаченности героев в команде.

# Командные показатели

- Для каждого индивидуального признака из [level, xp, gold, lh, kills, deaths, items] посчитаем следующие командные показатели:

- + минимальное и максимальное значение
- + сумму, среднее, медиану и дисперсию значений

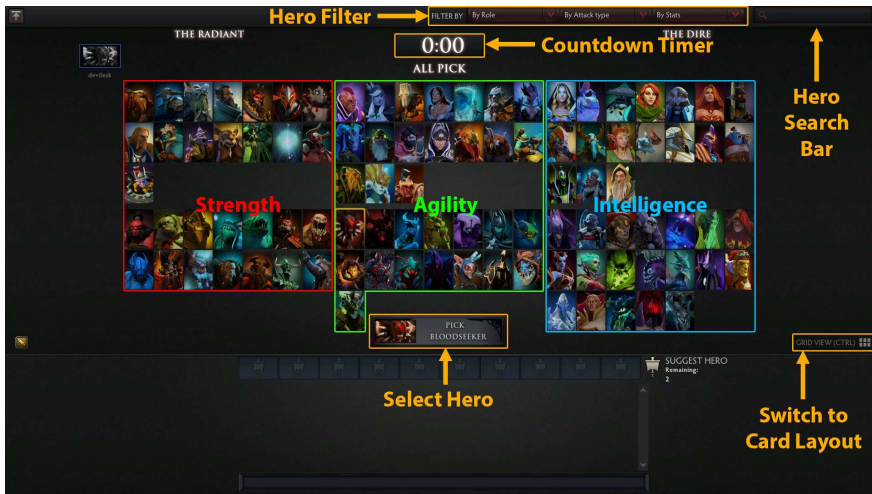
- Будем использовать каждый из признаков [first\_ward\_time, bottle\_time, courier\_time, flying\_courier\_time, tpscroll\_count, boots\_count, ward\_observer\_count, ward\_sentry\_count] как командный показатель

# Дополнительные данные

- У каждого героя есть свои показатели: сила, ловкость, интеллект, типа героя по этим показателям, прирост показателей на уровень и много чего другого
- Все это можно аналогично показателям в csv агрегировать на командные показатели



# Дополнительные данные
























# Дополнительные данные

Скачиваем [отсюда](#) дополнительные данные

- Придется повозиться с соответствием имён в скаченных данных и именах в файле `heroes.csv` — чтобы провести правильное соответствие между номерами героев и характеристиками скаченных героев

## Дополнительные данные

ГЕРОЙ	А	СИЛ	СИЛ+	ЛОВ	ЛОВ+	ИНТ	ИНТ+	О	О+	УР 25	ПЕР	БР	УРОН (МИН)	УРОН (МАКС)	ДАЛЬН	БВА	АНИМ	ЭПА	ОБЗ-Д	ОБЗ-Н	ПОВ
 Abaddon	🔴	23	2.7	17	1.5	21	2	61	6.2	209.8	310	1.38	55	65	150	1.7	0.56	0.41	1800	800	0.6
 Alchemist	🔴	25	1.8	11	1.2	25	1.8	61	4.8	176.2	295	1.54	49	58	150	1.7	0.35	0.65	1800	800	0.6
 Ancient Apparition	🟡	18	1.4	20	2.2	25	2.6	63	6.2	211.8	295	1.8	44	54	600	1.7	0.45	0.3	1800	800	0.6
 Anti-Mage	🟢	22	1.2	22	2.8	15	1.8	59	5.8	198.2	315	2.08	49	53	150	1.45	0.3	0.6	1800	800	0.5
 Arc Warden	🟢	24	2.3	15	1.8	24	2.6	63	6.7	223.8	285	0.1	44	54	625	1.7	0.3	0.7	1800	800	0.4
 Axe	🔴	25	2.5	20	2.2	18	1.6	63	6.3	214.2	290	1.8	49	53	150	1.7	0.5	0.5	1800	800	0.6
 Bane	🟡	22	2.1	22	2.1	22	2.1	66	6.3	217.2	310	4.08	55	61	400	1.7	0.3	0.7	1800	800	0.6
 Batrider	🟡	23	2.4	15	1.5	24	2.5	62	6.4	215.6	290	2.1	38	42	375	1.7	0.3	0.54	1200	800	1
 Beastmaster	🔴	23	2.2	18	1.6	16	1.9	57	5.7	193.8	310	4.52	64	68	150	1.7	0.3	0.7	1800	800	0.4
 Bloodseeker	🟢	23	2.4	24	3	18	1.7	65	7.1	235.4	290	3.36	53	59	150	1.7	0.43	0.74	1800	800	0.5
 Bounty Hunter	🟢	17	1.8	21	3	19	2	57	6.8	220.2	315	5.94	45	59	150	1.7	0.59	0.59	1800	1000	0.6
 Brewmaster	🔴	23	2.9	22	1.95	14	1.25	59	6.1	205.4	300	2.08	52	59	150	1.7	0.35	0.65	1800	800	0.6
 Bristleback	🔴	22	2.2	17	1.8	14	2.8	53	6.8	216.2	290	3.38	48	58	150	1.8	0.3	0.3	1800	800	1
 Broodmother	🟢	17	2.5	18	2.2	18	2	53	6.7	213.8	295	2.52	44	50	150	1.7	0.4	0.5	1800	800	0.5
 Centaur Warrunner	🔴	23	4	15	2	15	1.6	53	7.6	235.4	300	1.1	55	57	150	1.7	0.3	0.3	1800	800	0.5
 Chaos Knight	🔴	20	2.9	14	2.1	16	1.2	50	6.2	198.8	325	3.96	49	79	150	1.7	0.5	0.5	1800	800	0.5
 Chen	🟡	23	1.5	15	2.1	21	2.8	59	6.4	212.6	300	1.1	43	53	600	1.7	0.5	0.5	1800	800	0.6
 Clinkz	🟢	15	1.6	22	3.3	16	1.55	53	6.45	207.8	300	2.08	37	43	640	1.7	0.7	0.3	1800	800	0.4
 Clockwerk	🔴	24	2.9	13	2.3	17	1.3	54	6.5	210	315	1.82	55	57	150	1.7	0.33	0.64	1800	800	0.6
 Crystal Maiden	🟡	16	1.7	16	1.6	16	2.9	48	6.2	196.8	280	1.24	35	41	600	1.7	0.55	0	1800	800	0.5
 Dark Seer	🟡	22	2.3	12	1.2	25	2.7	59	6.2	207.8	300	6.68	56	62	150	1.7	0.59	0.58	1800	800	0.6

# Я сделаю

- Есть две фракции героев: светлые и темные. Логично было предположить, что светлые выбирают только из светлых, а темные — только из темных



Но не всегда стоит доверяться логике :) Как выяснилось на последующем командном этапе, это предположение оказалось неверным.

# Синергия

Так как одни герои действуют эффективно с/против других, то хочется учитывать попарное взаимодействие (синергию):

$$s_{ij} = \begin{cases} 1, & \text{если } i\text{-ый герой играет за radiant} \\ -1, & \text{если } i\text{-ый герой играет за dire} \end{cases}$$

- Будем это все хранить в формате разреженной матрицы, так как попарных признаков много (вкуче с количеством сэмплов)

Удивительно, но даже после такого бреда скор поднялся ;)

# Нормализация данных и модель

- Поскейлим с помощью `StandardScaler` из `sklearn` все данные, кроме синергии.
- Кросс-валидируем логистическую регрессию с L2-регуляризацией, выбираем коэффициент при регуляризации  $C$ , дающий наилучший скор (метрика `logloss`)

## Итог

- 0.57539 на пабlike (2 место на пабlike)
- 0.58280 на всех данных (1 место на всех данных)

# Предсказание вероятности победы Radiant в Dota2

Командный этап

Липкина Анна (317 группа)  
Думбай Алексей (317 группа)  
Гетоева Аида (517 группа)

25 ноября 2016 г.

# Diff

- Исправлена синергия (см. [презентацию](#)) (Улучшение сора на кросс-валидации и паблице на 0.003)
- Добавлены фичи из данных Лёши



# Сработавшие идеи

Идеи, которые улучшили результат:

- + Мешок предметов.
- + Разности средних значений, деленные на корень из суммы квадратов.
- + То же, только для суммы квадратов и кубов.
- + Отношение убийств команды к смертям.
- ++ Разность количества уничтоженных башен командами.

# Не сработавшие идеи

Идеи, которые ухудшили результат или не были закончены:

- Добавление количества выкупов героев по командам и их командный показатель
- Добавление убийств Рошана
- ? Анализ информации о конце матча на тестовой выборке
- ? Стекинг моделей регрессии с разной регуляризацией

# Итог

- Модель: логистическая регрессия с L2-регуляризацией

## Итог

- 0.56915 на пабlike (1 место на пабlike)
- 0.57918 на всех данных (1 место на всех данных)