Групповой проект по анализу характеристик игроков, участвовавших в профессиональных матчах по Dota 2

Команда №

Вступление

Привет!

Данный групповой проект посвящен реализации различных этапов анализа данных, с которыми вы успели ознакомиться на лекциях и семинарах. Датасет, с которым вы будете работать, содержит игровые характеристики участников профессиональных матчей по игре Dota 2.

Dota 2 - это многопользовательская онлайн-игра в жанре MOBA (Multiplayer Online Battle Arena), разработанная и выпущенная компанией Valve. Игра состоит из двух команд, каждая из которых включает по пять игроков. Каждая команда занимает свою базу на карте, и главная цель игры — разрушить главное здание на базе противника, известное как Ancient (часто называемое в ру-сегменте “троном”), защищая при этом своё. Игроки выбирают уникальных героев, каждый из которых обладает своими способностями и ролями на поле боя, и сражаются, зарабатывая золото и опыт для усиления своих персонажей.

Датасет включает различные характеристики игроков, краткое описание которых представлено ниже. Основной переменной, на которую будет направлен анализ, является “GPM” (Gold Per Minute), в датасете это “gold_per_min”, что переводится как “золото в минуту”. Этот показатель отражает, сколько золота в среднем зарабатывает игрок за минуту игрового времени, и является важным индикатором эффективности игрока (однако по ходу выполнения проекта вы будете обращать внимание и на другие показатели).

Однако сначала - условия проекта.

Условия и формат сдачи

Данный проект выполняется в командах до 5 человек (проект может выполнять и один человек при желании). Как только команда сформирована, необходимо вписать себя вот в эту таблицу:

→ Ссылка на таблицу ←

Оценки за задание будут выставлены там же.

Для сдачи проекта вы отправляете по ссылке ниже два файла:

- заполненный вами .Rmd файл, созданный из текущего, который содержит весь ваш исходный код и ответы
- созданный (скниченный/knitted) .pdf файл, скомпилированный из текущего .Rmd файла после выполнения работы. PDF-файл **не должен** содержать чанки кода (вы можете его скрыть). Вы можете вставлять визуализации, если считаете, что это контекстно релевантно.

В данной форме вы также указываете информацию о своей команде. **Оба файла отправляет один человек от команды**. В случае не единственной отправки от одной команды засчитываться будет последняя.

→ Ссылка на форму для сдачи группового проекта ←

В конце проекта есть несколько индивидуальных вопросов. На них каждый член команды отвечает отдельно (не внутри командного документа) и отправляет ответы по ссылке ниже:

→ Ссылка на форму с индивидуальными вопросами ←

ИТОГО

Каждая команда должна:

- занести участников своей команды в гугл-таблицу по ссылке
- отправить командный .Rmd, командный .pdf изаполнить форму по ссылке. Форма заполняется одним человеком от команды.

Каждый член команды должен:

- заполнить индивидуальную форму по ссылке

Мягкий дедлайн сдачи проекта: 02.06.2024 23:59

Жёсткий дедлайн сдачи проекта: 09.06.2024 23:59

После сдачи проекта после мягкого дедлайна каждые сутки просрочки накладывают штраф в $\frac{10}{7} \approx 1,429$ баллов. Сдача проекта 09.06 ведет к потере 10 баллов. После жёсткого дедлайна задание не принимается.

Формула оценивания

Задание состоит из:

- 24 вопросов основной части и 1 бонусного вопроса в основной части
- 1 вопроса в блоке **Выводы**
- 2 индивидуальных вопросов

Максимальный балл за корректные ответы на все вопросы - 20 (не считая бонусного вопроса).
Каждому блоку задания присвоена своя сложность:

Вопросы	Сложность
Q1-Q8	1
Q9-Q16	1.5
Q17-Q20	2.5
Q22-Q25	3
Q26	1
Q27-Q28	0.5

Вопрос Q21 является бонусным и входит как доп. балл со сложностью, равной $\frac{2.5}{4}$ (как если бы это было задание из категории Q17-Q20).
Одна единица сложности задания пересчитывается в баллы следующим образом:

$$\frac{20}{1 + 1.5 + 2.5 + 3 + 1 + 0.5} = \frac{20}{9.5} \approx 2,105$$

Так, весь блок Q1-Q8 весит 2.105 баллов, а любой вопрос из этого блока весит $\frac{2.105}{8} \approx 0,263$ балла.
Вопрос Q21 весит $\frac{2.105 \times 2.5}{4} \approx 1,316$ балла. Все округления происходят уже при подсчете финального балла за всё задание (после суммирования с Q21). При наборе 20 баллов бонусные баллы могут быть распределены в сторону остальных элементов контроля (пересчет в баллы других элементов контроля реализуется в зависимости элемента контроля).

Если задание имеет подпункты, каждый подпункт имеет вес вопроса, делённый на число подпунктов. Так, вопрос Q5 имеет два подпункта, каждый из которых имеет сложность $\frac{\text{Q5}}{2}$.

Описание датасета

- match_id. Уникальный ID матча.
- player_slot. ID слота игрока.
- account_id. ID аккаунта игрока.
- additional_units. Если у героя есть возможность иметь призванное существо, которое ведёт себя как герой, переменная будет содержать информацию о таком существе.
- assists. Количество раз игрок принимал участие в убийстве вражеского героя.
- camps_stacked. Количество стаков нейтральный лагерей данным игроком.
- creeps_stacked. Количество нейтральных крипов, “накопленных” в нейтральных лагерях благодаря стакам, выполненным данным героем.
- deaths. Количество смертей.
- firstblood_claimed. Индикатор первого убийства в матче.
- gold_per_min. GPM, среднее количество золота в минуту.
- gold_spent. Количество потраченного золота за матч.
- hero_healing. Количество здоровья, которое данный игрок восстановил союзным героям.
- hero_id. ID героя.
- kills. Количество убийств.
- kills_log. Лог убийств; содержит информацию о времени совершения убийства, а также идентификатоы убитых героев.
- last_hits. Количество добиваний (нанесений летального урона) нейтральных крипов данным героем.
- level. Уровень данного героя на момент конца матча.
- obs_placed. Количество тотемов наблюдений, размещенных на карте данным игроком.
- teamfight_participation. Степень участия в сражениях с вражескими героями, [0; 1]
- radiant_win. Индикатор победы сил света.
- start_time. Дата начала матча.
- duration. Длительность матча, в секундах.
- game_mode. Индикатор игрового режима.
- patch. Идентификатор патча, являющимся актуальным на момент проведения матча.
- isRadiant. Индикатор принадлежности данного игрока силам света в данным матча.
- win. Индикатор победы.
- lose. Индикатор поражения.
- kda. KDA, kills/deaths/assists (про это чуть ниже)
- neutral_kills. Количество нейтральных крипов, убитых данным героем.
- lane_kills. Количество убитых крипов вражеской команды.
- observer_kills. Количество уничтоженных вражеских тотемов наблюдения.
- ancient_kills. Количество убитых древних нейтральных крипов.
- lane. Идентификатор линии, на которой находился в начале матча данный игрок.
- region. Регион, в котором проводился матч.

EDA

Часть 1

В данной части вам предстоит провести разведывательный анализ предоставленного датасета.

`match_id` отвечает за уникальный номер матча, в каждом из которых соревнуются две команды - силы света (Radiant) и силы тьмы (Dire)

Оставьте только одну переменную для идентификации команды (`isRadiant`) и сделайте ее бинарной. Удалите переменные `player_slot` и `account_id`.
Переменную `additional_units` стоит удалить из-за превалирования пропусков.
Проходясь по остальным переменным, мы можем удалить следующие:

- `patch`. Датасет сформирован так, что за рассматриваемый период игра не претерпевала серьёзных изменений, а патчи были минорными, так что в основной части задания от вас не требуется производить анализ каждого матча в отдельности
- `region`. Пока удаляем, в основной части нам это не пригодится
- `game_mode`. Большинство матчей проходило на профессиональных турнирах. Такие турниры проходят в режиме "Captain's mode" - в таком режиме один участник команды выбирает героев для всех остальных. Данный режим закодирован здесь как "2". Оставьте только те матчи, которые проходили в таком режиме (`game_mode` имеет значение **2**). Убедитесь, что каждый уникальный матч встречается в датасете 10 раз (т.е. для каждого матча есть информация по всем 10 участникам). Если остались матчи с информацией не по всем 10 участникам, удалите их
- `lose`. У нас уже есть индикатор победы (`win`).
- `radiant_win`. У нас уже есть индикаторы `isRadiant` и `win`

Q1 Сколько уникальных матчей осталось в датасете?

Ваш ответ здесь

В игре существует множество прямых способов получения золота (и отсюда - увеличения показателя “Количество золота в минуту”, GPM). Из представленных в датасете:

- участие в убийстве вражеского героя (`assists`);
- убийство вражеского героя (`kills`). В частности, за первое убийство на карте дают бонусное золото тому, кто нанёс финальный удар (переменная `firstblood_claimed`);
- переменная `observer_kills`[1] отвечает за уничтожение вражеских тотемов наблюдения и вносит вклад в переменную GPM.
- `camps_stacked`[2] - показатель того, сколько раз герой “настакал” таких лагерей за матч, в то время как `creeps_stacked` - показатель общего числа крипов, которых герой настакал. Например, в видео выше герой выводит из лагеря сатиров, и в лагере появляются 3 тролля. В таком случае переменная `camps_stacked` будет равна 1 (появление в лагере крипов благодаря игроку осуществлено один раз), а `creeps_stacked` будет равна 3 (так как благодаря действиям игрока появилось 3 новых крипа).
- `last_hits` - количество добитых крипов. Складывается из `neutral_kills` (нейтральных крипов) и `lane_kills` (вражеских крипов, идущих по одной из трех игровых линий). В свою очередь `lane_kills` складывается из `ancient_kills` (древних нейтральных крипов) и `nonancient_kills` (обычных нейтральных крипов). Древние крипы - те же нейтральные, но больше и сильнее.

Количество смертей (`deaths`) не влияет на показатель GPM, т.к. последний отражает количество заработанного золота и не учитывает потери. Будучи мёртвым, много не

[1] **Пояснение про тотем наблюдения (observer ward)**.
У каждого героя на карте есть некоторый радиус обзора вокруг себя, однако при отсутствии дружественных юнитов или построек за его пределами герой не видит ничего (видит туман). Предмет observer ward - тотем, который ставится на землю/низменность/возвышенность и даёт широкий обзор вокруг себя. Observer ward невидим, однако с помощью другого тотема, sentry ward, его можно обнаружить и разломать. В контексте задачи для нас важны две вещи:

1. Чем больше дружественных тотемов на карте (поставленных в нужное место), тем больше информации есть у команды.
2. За уничтожение вражеского observer ward’а герой, уничтоживший тотем, получает золото.

[2] **Пояснение про лагери нейтральных крипов**
В доте сущности, не являющиеся героями, называются *крипами*. В игре есть три вида крипов по принадлежности к команде: дружественные, вражеские и нейтральные. Нейтральные крипы находятся в статично закрепленной за ними территории, называемой лагерем. Такие лагери разбросаны по всей игровой карте и связаны с механикой, по кальке называемой “стакинг”. Если в текущий момент на игровых часах время в секундах кратно 60, и в каком-то лагере нейтральных крипов нет (скорее всего их кто-то убил), они появятся там снова (заспавнятся).
Механика стакинга состоит в том, чтобы на подходе к полной минуте (например, 1:55 на часах) ударить одного из таких крипов, увести его за собой (и таким образом вывести за пределы лагеря) и стриггерить появление новых внутри (так как игра будет считать, что крипов внутри лагеря нет). Если сложно, вот иллюстрирующая картинка и видео.
Убивая крипов на карте (вражеских и нейтральных), игрок получает деньги. Также игрок получает деньги, если кто-то убьет крипов в стакнутом им лагере.

заработаешь, однако информацию, связанную с этой переменной, мы всё равно будем использовать.

Существуют и показатели, которые косвенно могут влиять на показатель GPM (однако в этом ещё предстоит убедиться). Одна из них - `obs_placed`

Переменная `obs_placed` отражает количество тотемов наблюдения, поставленных одним игроком в течение матча. Максимальное время жизни тотема - 6 минут, после этого он автоматически разламывается, и золото не достается никому. Косвенное влияние данной перемены на GPM может быть в том, что, имея достаточную информацию о положении вражеских игроков на карте, команда может более разумно планировать свои действия и оптимизировать получение золота (например, начав драку более подготовленными или вовремя её избежав).

Поработаем с оставшимися переменными.

Частью выполнения задания является построение регрессионной модели для предсказания переменной `gold_per_minute`.

Q2 Как вы думаете, почему для этого нам нужно удалить переменную `gold_spent` из датасета? (ответьте на вопрос и удалите её)

Ваш ответ здесь

Давайте посмотрим на самих героев и их характеристики.
Загрузите датасет **heroes.csv**

У героев есть два типа атаки - ближний и дальний. Также у героев есть четыре базовых атрибута:

- Ловкость (agi)
- Сила (str)
- Интеллект (int)
- Универсальный атрибут (all)

Как правило именно базовый атрибут героя возрастает сильнее остальных по мере получения следующих уровней.
Добавьте из **heroes.csv** переменные `primary_attr` и `attack_type` в уже имеющийся датасет, объединив их по ключу *hero_id*.

Q3 Герой с каким ID имеет больше всего GPM? Если таких героев несколько, укажите ID всех через запятую.

Ваш ответ здесь

Q4 Герой с каким ID имеет наибольший показатель GPM среди всех “ловкачей”?

Ваш ответ здесь

Q5.1 У каких героев выше средний показатель GPM - у героев с ближним типом атаки (melee) или дальним (ranged)?

Ваш ответ здесь

Q5.2 Что лучше использовать в данном случае - медиану или среднее? Сильно ли отличаются средние значения от медианных? Продолжим работу с существующими переменными.

Ваш ответ здесь

В датасете есть переменная `kills_log`, представляющая из себя лист json-like объектов, находящийся в строковом формате. Каждый json имеет следующее устройство:

{“time”: *time_value*, “key”: *key_value*}, где `time_value` - время, в которое произошло убийство (с начала матча в секундах), а `key_value` - код убитого героя.

Обработайте каждую такую строку, достав из неё временнЫе отметки убийств и добавьте в датасет новую переменную `median_free_time`, которая определяется следующим образом:

$$\text{median}(\text{time\_value}_1, \text{time\_value}_2 - \text{time\_value}_1, \dots, \text{time\_value}_n - \text{time\_value}_{n-1})$$

Иными словами, нам интересен медианный промежуток между убийствами. Это даёт дополнительную информацию, которой мы бы смогли оценить интенсивность сражений в течение игры, а не только их количество.

Некоторые значения переменной `kills_log` отсутствуют (NA). Заполните их средним значением переменной `median_free_time`, которую получилось посчитать для `kills_log` без пропусков. Пусть это среднее значение будет называться `mean_median_free_time`.

Также некоторые значения переменной kills_log представляют из себя пустую строку (“[]”). Убедитесь в том, что это означает отсутствие убийств данным героем вообще и:

- если это так (убийства отствуют), для таких наблюдений присвойте переменной `median_free_time` значение, равное значению переменной `duration`,
- если это не так (убийства присутствуют), в качестве значения для `median_free_time` присвойте значение, равное `mean_median_free_time`.

Q6 Чему равно значение `mean_median_free_time`?

Ваш ответ здесь

Удалите переменную `kills_log`.
Мы почти закончили с созданием новых переменных!
Переменная `start_time` содержит дату в формате “mm/dd/yyy”. Переведите дату в unix формат.
Переменная `kda` (kills/deaths/assists) по смыслу должна означать kda ratio, которое формульно вычисляется как:

$$\text{kda\_ratio} = \begin{cases} \frac{\text{kills}+\text{assists}}{\text{deaths}}, & \text{deaths} \neq 0 \\ \text{kills} + \text{assists}, & \text{deaths} = 1 \end{cases}$$

Пересчитайте kda по этой формуле, создайте для этого переменную `kda_ratio` и удалите переменную `kda`.

Q7 Чему равно медианное значение переменной `kda_ratio`?

Ваш ответ здесь

Поработаем с переменной `level`.
Количество опыта, необходимое для достижения каждого последующего уровня, неодинаково. В частности, можно заметить, что функция общего количества опыта от уровня героя имеет почти экспоненциальный рост!
Загрузите json-файл **level_exp.json**, содержащий словарь в формате {*level*: *experience*}.
Создайте в вашем датасете переменную `exp`, равную значению *experience* из json-файла, смэтчив json и датасет по ключу *level*.
Переменную `level` в датасете удалите.

Осталось последнее преобразование, ведущее к созданию новой переменной, и это преобразование связано с переменной `lane`.
Удалите все матчи, для которых в датасете есть наблюдения с пропусками в переменной `lane` вида NA (т.е. если в каком-то матче есть игрок, для которого `lane` неопределен (NA), удалите все 10 наблюдений, относящиеся к этому матчу).

Теперь создайте три новые бинарные переменные - `lane_1`, `lane_2`, `lane_3`, преобразовав переменную `lane`. Если значение `lane` равно 1, `lane_1` будет равно 1, а `lane_2` и `lane_3` - 0 (для `lane` = 2 и `lane` = 3 логика та же).

Переменную `lane` удалите.
По аналогии поступите с переменными `primary_attr` и `attack_type`.

Создайте новые бинарные переменные из `primary_attr`:

- `str_attr`
- `agi_attr`
- `int_attr`
- `all_attr`

И новые бинарные переменные из `attack_type`:

- `melee_attack`
- `ranged_attack`

Переменные `primary_attr` и `attack_type` удалите.

Убедитесь, что в вашем датасете на данный момент **32** переменные:

- duration
- start_time
- assists
- deaths
- kills
- teamfight_participation
- camps_stacked
- creeps_stacked
- last_hits
- lane_kills
- firstblood_claimed
- obs_placed
- observer_kills
- neutral_kills
- ancient_kills
- nonancient_kills
- isRadiant
- win
- median_free_time
- kda_ratio
- exp
- lane_1
- lane_2
- lane 3
- str_attr
- agi_attr
- int_attr
- all_attr
- melee_attack
- ranged_attack
- hero_healing
- gold_per_min

Если это не так, найдите ошибки и исправьте перед тем как приступить ко 2 части блока **EDA**

Часть 2

В этой части блока **EDA** мы будем преимущественно очищать значения и смотреть на распределения переменных Переменная `teamfight_participation` должна принимать значения в интервале [0; 1]

Замените значения teamfight_participation > 1 на 1.

Q8 Чему равно среднее значение `teamfight_participation` после удаления выбросов? А медианное? Какое значение вы бы скорее использовали в качестве меры центральной тенденции и почему?

Ваш ответ здесь

Переменная `last_hits` также посчитана неправильно. Замените её значение на сумму

$$neutral\_kills + lane\_kills$$

Создайте переменную `nonancient_kills`, равную

$$neutral\_kills - ancient\_kills$$

а переменную `neutral_kills` удалите.
Переменная `hero_healing` имеет очень много нулей, та же проблема существует у переменной `exp`.

Создайте новые переменные-индикаторы (бинарные переменные), `healing_zero` и `exp_zero`, принимающие значение 1, если `hero_healing` и `exp` (соответственно) принимают значение 0. В остальных случаях `healing_zero` и `exp_zero` равны 0.

EDA (продолжение) и корреляционный анализ

Теперь давайте посмотрим на переменные `assists`, `deaths` и `kills`

Часто бывает полезно приблизить переменные, чье распределение лог-нормальное, к нормальному, перед тем как использовать их в качестве предикторов в линейных моделях (коей является линейная регрессия), поскольку это приближает линейность.

Один из наиболее часто используемых способов - взятие логарифма Однако логарифм "сжимает" разброс значений, а также всё-таки меняет форму распределения, так что это не всегда "silver bullet".

Альтернативный вариант - взятие корня, однако у этого варианта есть свои нюансы:

- корень четной степени не получится использовать при наличии отрицательных значений;
- при сильной скошенности корень скорее всего не будет работать так же хорошо, как логарифм;
- логарифм проще интерпретируется.

В частности, рассмотрите три случая ниже.

Первый

$$ln(Y) = \beta0 + \beta1 \times X$$

Параметр β1 показывает процентное изменение Y при увеличении X на 1. Поскольку ln(Y) - линейная функция от X, β1 интерпретируется как приблизительное процентное изменение в Y. В частности, увеличение X на 1 ведёт к увеличению Y на exp(β1).

Например, β1=0.5. Тогда увеличение X на 1 единицу ведёт к увеличению Y на $exp(0.5) = 1.648$, что по смыслу равно росту Y на 65% $[(1.648 - 1) \times 100\%]$

Второй

$$Y = \beta0 + \beta1 \times ln(X)$$

Параметр β1 показывает абсолютное изменение в Y при 1%-ом изменении в X. Поскольку 1%-е увеличении X ведет к изменению $ln(X)$ на $ln(1.01) \approx 0.01$, β1 показывает изменение в Y на это увеличение.

Например, β1=0.2. Тогда увеличение X на 1% ведет к увеличению $ln(X)$ на 0.01, что ведет к увеличению Y на $0.01 \times \beta1 = 0.002$.

Третий

$$ln(Y) = \beta0 + \beta1 \times ln(X)$$

Параметр β1 показывает эластичность Y относительно X, то есть процентное изменение в Y при процентном изменении в X. Эластичность означает, что β1 можно интерпретировать как процентное изменение в Y при увеличении X на 1%.

Например, β1=0.3. Тогда 1%-е ведет к увеличению $ln(X)$ на 0.01, а $ln(Y)$ растёт на $0.3 \times 0.01 = 0,003$. В таком случае Y растет на $exp(0.003) \approx 1,003$, т.е. на 3%

Q9.1 Посмотрите на корреляцию между переменной `kills` и `gold_per_min`, значима ли она при α=0.05? Чему равно p-value?

Ваш ответ здесь

Q9.2 Возьмите натуральный логарифм от `kills` вида ln(1+kills) и посмотрите на корреляцию этой переменной с `gold_per_min`. Как изменилась корреляция? Осталась ли она значимой?

Ваш ответ здесь

Вопрос для себя: почему мы применяем трансформацию вида ln(1+kills), а не ln(kills)?

Посмотрите на переменные `hero_healing` и `exp`
Для каждой переменной отдельно рассмотрите подмножества датасета, в которых healing_zero $=$ 0 и exp_zero` $=$ 0 соответственно. В каждом таком подмножестве посмотрите на форму распределения этих переменных.

Q10.1 Распределены ли они нормально?

Ваш ответ здесь

Если их распределение далеко от нормального и скошено вправо, попробуйте логнормировать каждое или взять корень (попробуйте степени вплоть до 5).

Q10.2 Получилось ли приблизить распределение каждой переменной к нормальному? Если да, что помогло лучше?

Ваш ответ здесь

На этом шаге новых переменных создавать не нужно.

Q11.1 Попробуйте найти переменную (или переменные), разбиение по которой (которым) датасета на группы избавит `gold_per_min` от бимодальности. Что это за переменная/переменные?

Ваш ответ здесь

Q11.2 Попробуйте найти переменную (или переменные), разбиение по которой (которым) датасета на группы избавит `last_hits` от бимодальности. Что это за переменная/переменные?

Ваш ответ здесь

Постройте матрицу корреляций континуальных переменных.

Q12.1 Какие 3 переменные лучше всего коррелируют с `gold_per_min`?

Ваш ответ здесь

Q12.2 Являются ли эти корреляции статистически значимы при $\alpha=0.05$? Чему равно p-value?

Ваш ответ здесь

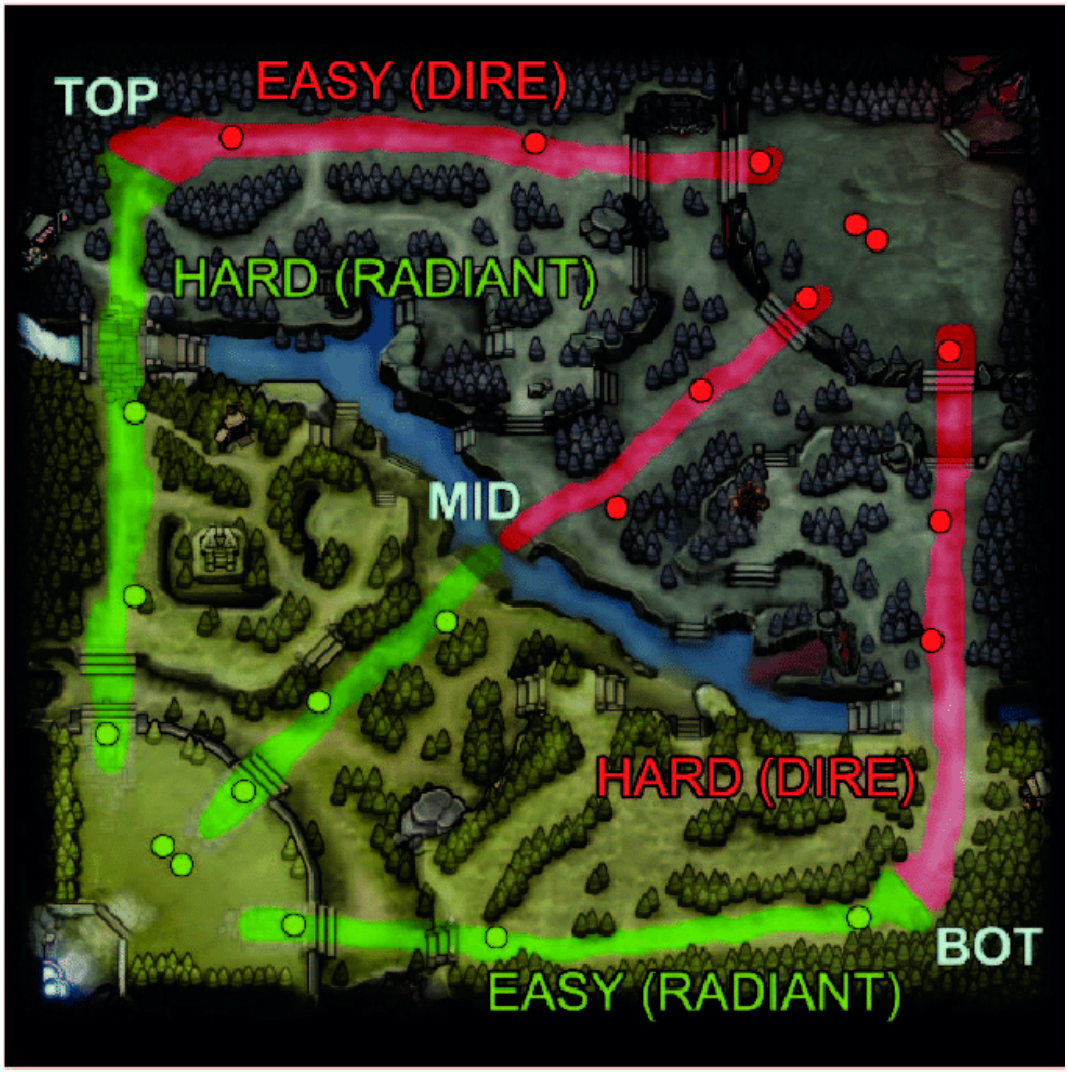
Возможно, некоторые из этих 3 переменных сильно связаны друг с другом, и тогда среди этих топ-3 сильно отличающихся друг от друга по смыслу переменных на самом деле меньше. Проверьте, не коррелируют ли между собой какие-то из переменных, вошедших в ваш топ-3.

Q12.3 Обнаружили ли вы статистически значимые (при $\alpha=0.05$) корреляции между какими-либо переменными из этого топа? Если да, как вы думаете, с чем это может быть связано?

Ваш ответ здесь

Теперь давайте посмотрим на подгруппы!

В датасете есть переменные `lane_1`, `lane_2` и `lane_3` (созданные вами из переменной `lane`). Игровая карта устроена следующим образом:



Каждая из команд имеет прямой доступ к базе команды противника через три основные линии (центральную, самую короткую, и две боковые). Герои на карте зачастую располагаются так, что на центральной линии встречаются по одному герою от каждой команды, а на боковых - по два. При таком расположении центральные герои имеют бОльшую зависимость от индивидуальных навыков, быстрее получают опыт (experience), быстрее получают золото.
Возможно, для таких героев топ-3 корреляций с `gold_per_min` будет выглядеть иначе?
Разбейте датасет на две части по переменной `lane_2` и постройте матрицу корреляций для каждой из получившихся подгрупп.

Q13.1 Какие 3 переменные лучше всего коррелируют с `gold_per_min` в датасете с героями, стоявшими на центральной линии (`lane_2` = 1)? Значимы ли эти корреляции при α=0.05?

Чему равно p-value?

Ваш ответ здесь

Q13.2 Какие 3 переменные лучше всего коррелируют с `gold_per_min` в датасете с героями, стоявшими на боковых линиях (`lane_2` = 0)? Значимы ли эти корреляции при α=0.05? Чему равно p-value?

Ваш ответ здесь

Скорее всего герои, не участвующие в сражениях вместе с командой, занимаются чем-то другим, а значит их источники заработка золота могут отличаться от имеющихся у героев, ведущие эти сражения часто.

Q13.3 Отличаются ли результаты в подгрупах от того, что вы получили для всего датасета целиком? Как вы думаете, с чем это связано?

Ваш ответ здесь

Разбейте датасет на две части по медиане переменной `teamfight_participation`.

Q14.1 Какие 3 переменные лучше всего коррелируют с `gold_per_min` в датасете с героями, имеющими значение `teamfight_participation` < медианы? Значимы ли эти корреляции при α=0.05? Чему равно p-value?

Ваш ответ здесь

Q14.2 Какие 3 переменные лучше всего коррелируют с `gold_per_min` в датасете с героями, имеющими значение `teamfight_participation` >= медианы? Значимы ли эти корреляции при α=0.05? Чему равно p-value?

Ваш ответ здесь

Q14.3 Отличаются ли результаты в подгрупах от того, что вы получили для всего датасета целиком? Как вы думаете, с чем это связано? Возможно, есть какое-то отличие у факторов, влияющих на `gold_per_min` в зависимости от типа атаки?

Ваш ответ здесь

Разбейте датасет на две части по типу атаки (одна подгруппа - ranged_attack $= 1$, другая - melee_attack$` = 1$)

Q15.1 Какие 3 переменные лучше всего коррелируют с `gold_per_min` в датасете с героями, имеющими ближний тип атаки? Значимы ли эти корреляции при α=0.05? Чему равно p-value?

Ваш ответ здесь

Q15.2 Какие 3 переменные лучше всего коррелируют с `gold_per_min` в датасете с героями, имеющими дальний тип атаки? Значимы ли эти корреляции при α=0.05? Чему равно p-value?

Ваш ответ здесь

Q15.3 Отличаются ли результаты в подгрупах от того, что вы получили для всего датасета целиком? Как вы думаете, с чем это связано?

Ваш ответ здесь

Q16.1 - Q16.3 Попробуйте проделать эти же шаги, разбивая датасет на подгруппы дальше.

Ваш ответ здесь

Например, внутри группы героев, стоявших на центральной линии, сравните подгруппы героев с ближним и дальним типами атак (вы можете выбрать другие подгруппы).
На вопросы 17.1 и 17.2 ответы должны так же содержать информацию о топ-3 наиболее скоррелированных переменных с нашей целевой (`gold_per_min`), а также информацию об уровне значимости (p-value).

Q16.4 Чем был обусловлен для вас выбор новых подгрупп?

Ваш ответ здесь

ВАЖНО!
Обратите внимание на количество наблюдений в получившихся подгруппах, т.к. их может оказаться крайне мало. Например, среди героев ближнего боя (melee_attack $= 1$) нет ни одного героя, чьим основным атрибутом является интеллект (int_attr $= 1$).

Статистические тесты

Данный блок посвещён проведению статистических тестов для проверки гипотез.

Одной из часто обсуждаемых тем (по крайней мере так было долгое время) в контексте игры Dota 2 является несимметричность игровой карты, а также разные углы обзора в зависимости от команды - так, например, герои стороны света смотрят на карту как бы снизу-вверх, а стороны тьмы - сверху-вниз из-за расположения их баз.
Проведите двухвыборочный двухсторонний t-test для проверки следующей гипотезы (H0): вероятность победы не зависит от стороны, за которую играет команда (т.е. доля побед команд, игравших за сторону света, не отличается от доли побед команд, игравших за сторону тьмы). Обратите внимание на то, что единицами наблюдения в датасете являются герои, а не матчи, поэтому датасет необходимо предобработать.

В качестве параметра α возьмите 0.05.

Q17 Чему равно p-value? Удалось ли отклонить H0? Как вы думаете, почему вы получили такой результат?

Ваш ответ здесь

Теперь вернёмся к переменной, представляющей для нас наибольшей интерес, - `gold_per_min`
Проверьте следующую гипотезу (H0): средняя величина `gold_per_min` не отличается между двумя группами.

В качестве двух групп для сравнения возьмите:

- melee_attack $= 1$ vs ranged_attack $= 1$
- healing_zero $= 1$ vs healing_zero $= 0$

Параметр α возьмите 0.05.

Q18 В случае сравнения групп по типу атаки:

- Чему равно p-value?
- Удалось ли отклонить H0? Если да, в какой из двух рассмотренных групп показатель `gold_per_min` в среднем выше?
- Как вы думаете, почему вы получили такой результат?

  Ваш ответ здесь

Q19 В случае сравнения групп по наличию какого-либо лечения союзников:

- Чему равно p-value?
- Удалось ли отклонить H0? Если да, в какой из двух рассмотренных групп показатель `gold_per_min` в среднем выше?
- Как вы думаете, почему вы получили такой результат?

Ваш ответ здесь

Теперь давайте сравним показатель `gold_per_min` внутри трех групп: `lane_1`, `lane_2`, `lane_3`

В частности, нам интересно, одинаковое ли количество золота в минуту получают герои, расположенные на разных линиях (в **Q14** мы сравнивали `lane_2` с `lane_1` и `lane_3`, объединив последние две группы в одну). Можно было бы пойти по уже (надеемся) привычному пути t-тестов, однако есть подвох.

Пусть μ1 - средний `gold_per_min` для `lane_1` = 1, μ2 - средний `gold_per_min` для `lane_2` = 1 и т.д., а наша H0 следующая:

μ1 = μ2 = μ3 (разницы между группами нет). Пусть $\alpha = 0.05$

Для проверки такой гипотезы нам бы потребовалось проводить 3 парных теста (сравнивать μ1 с μ2, μ1 с μ3, μ2 с μ3), и это бы повлияло на вероятность совершения ошибки первого рода. Если бы мы положили $\alpha = 0.05$ для каждого из таких сравнений, вероятность совершить ошибку первого рода хотя бы одном тесте была равна ~14%. Один из частых способов борьбы с данной проблемой (при малом количестве попарных сравнений) - коррекция Бонферрони.

Суть метода состоит в том, чтобы при выборе уровня α для проверки основной гипотезы проводить попарные сравнения при уровне значимости не α, а α/m, где m - количество групп.
В нашем случае (3 группы) тест для каждой пары будет проводиться при уровне значимости $\alpha = 0.05/3 = 0,01667$. Если в случае каждого из трех тестов разницы между **каждой парой** не обнаруживается, то H0 не отклоняется при **α=0.05**. При этом если вы обнаружили разницу между двумя парами групп, такая разница будет статистически значима при α/3, поскольку это рассматривалось вне парадигмы нашей исходной нулевой гипотезы.

Возвращаясь к примеру с подсчетом вероятность совершения хотя бы одной ошибки первого рода, называемой также *familywise error rate* (FWER), равной ~14% ($1 - (1 - \alpha)^m = 1 - (0.95)^3$), давайте посмотрим, как меняется FWER после коррекции Бонферрони.

Вероятность совершения хотя бы одной ошибки равна $1 - (1 - \alpha/3)^3 = 1 - (0.983)^3 = 1 - 0,95083 = 0.04917 \approx 0.05$, что как раз равно нашей изначальной α!

Q20.1 Проведите парные двухсторонние t-тесты для проверки гипотезы о том, что между группами `lane_1`, `lane_2` и `lane_3` нет разницы в среднем `gold_per_min`. В качестве α возьмите 0.05. Отклонили ли вы H0?

Ваш ответ здесь

Q20.2 Чему равны p-value для каждой пары сравнений? Если H0 не была отклонена, между какими группами существует статистически значимая разница?

Ваш ответ здесь

На картинке выше (перед **Q13**) вы можете видеть подписи “easy” и “hard” около линий. Названия связаны с тем, с началом игры со стороны базы каждой команды вдоль каждой из линий движутся крипы сторон света и тьмы, маленькие миньоны, дерущиеся друг с другом, но способные также атаковать постройки и героев вражеских команд. Приведем пример со стороной света (с тьмой будет так же). Проходя по “лёгкой” линии стороны света, крипы сил света встречают крипов сил тьмы, которые идут навстречу по сложной линии стороны тьмы. Иными словами, лёгкая линия света соединена со сложной линией тьмы (и наоборот - сложная линия сил света соединена с лёгкой линией сил тьмы), однако откуда названия?

Дело в том, что крипы сил света, идущие по лёгкой линии, встречают крипов сил тьмы очень близко к защитной башне сил света (из-за несимметричности игровой карты), из-за чего герои сил света, находящиеся на легкой линии, могут убивать вражеских крипов, находясь под защитой своей башни, что делает их линию “лёгкой”, а вражескую - сложной. К сожалению, до конца непонятно, как именно закодированы игровые линии 1 и 3, то есть какая из них лёгкая, а какая - сложная.

Q20.3 Как вы думаете, с чем могут быть связаны полученные результаты сравнений подгрупп?

Ваш ответ здесь

Q21 (бонусное). Данное задание выполнять необязательно для получения максимального балла, однако за его выполнение вы можете получить дополнительные баллы.

Q21.1 Проведите парные двухсторонние t-тесты для проверки гипотезы о том, что между группами `str_attr`, `agi_attr`, `int_attr` и `all_attr` нет разницы в среднем `gold_per_min`. В качестве α возьмите 0.05. Отклонили ли вы H0?

Ваш ответ здесь

Q21.2 Чему равны p-value для каждой пары сравнений? Если H0 не была отклонена, между какими группами существует статистически значимая разница?

Ваш ответ здесь

Q21.3 Как вы думаете, с чем могут быть связаны полученные результаты сравнений подгрупп?

Ваш ответ здесь

Регрессия

Перейдём к последнему блоку! Постройте линейную регрессию с одной независимой переменной, имеющей наибольшую корреляцию с `gold_per_min`

Q22.1 Значим ли коэффициент beta при данной переменной (при α=0.05)? Чему равно p-value для этого коэффициента? Проинтерпретируйте коэффициент

Ваш ответ здесь

Q22.2 Чему равно значение R^2 для данной модели? Проинтерпретируйте значение R^2

Ваш ответ здесь

Q22.3 Является ли объясняющая переменная нормально распределенной? Если нет, попробуйте применить к ней логарифм или корень и сравнить результаты. Изменился ли R^2?

Ваш ответ здесь

Теперь попробуйте запустить линейную регрессию со всеми имеющимися переменными. Целевой переменной является `gold_per_min`.

Обратите внимание на то, что из датасета нужно удалить одну из категорий, полученных из :

- `lane` (удалите `lane_3`)
- `attack_type` (оставьте только `melee_attack`)
- `primary_attr` (удалите `all_attr`)

В датасете должны остаться следующие переменные среди независимых:

- duration
- start_time
- assists
- deaths
- kills
- teamfight_participation
- camps_stacked
- creeps_stacked
- last_hits
- lane_kills
- firstblood_claimed
- obs_placed
- observer_kills
- ancient_kills

- nonancient_kills
- isRadiant
- win
- median_free_time
- kda_ratio
- exp
- lane_1
- lane_2
- str_attr
- agi_attr
- int_attr
- melee_attack
- hero_healing
- healing_zero
- exp_zero

Q23.1 Какие коэффициенты (по абсолютной величине или знаку) кажутся вам наиболее неожиданными? Почему?

Ваш ответ здесь

Q23.2 Чему равно значение R^2 для данной модели? Возросло ли в сравнении с моделью с одной переменной? А adj. R^2?

Ваш ответ здесь

Создайте копию текущего датасета. Найдите среди ваших независимых переменных скошенные и попробуйте их логарифмировать (или взять корень) для приближения распределения к нормальному, запустите регрессию ещё раз.

Q24.1 Чему равно значение R^2 для данной модели? Возросло ли оно в сравнении с моделью с одной переменной? Возросло ли оно в сравнении с моделью без применения логарифма/корня? А adj. R^2?

Ваш ответ здесь

Q24.2 Дайте интерпретацию коэффициентам при переменных, к которым вы применили корень/логарифм

Ваш ответ здесь

Создайте ещё одну копию датасета из **Q24.X**. Стандартизируйте независимые континуальные

переменные в скопированном датасете. Постройте на этом датасете регрессию снова. Обратите внимание, что R^2 останется тем же, мы просто поменяли масштаб.

Теперь, когда переменные имеют одинаковый масштаб, мы можем сравнить коэффициенты между собой.

Q25.1 Какие 3 переменные вносят наибольший положительный вклад в GPM? Ожидаемо ли это?

Ваш ответ здесь

Q25.2 Какие 3 переменные вносят наибольший отрицательный вклад в GPM? Ожидаемо ли это?

Ваш ответ здесь

Выводы

Q26

- Какие выводы можно сделать выводы из проведенного вами анализа?

  Ваш ответ здесь

- Что удалось сделать, а что сделать не удалось?

  Ваш ответ здесь

- Как можно было бы улучшить вашу работу? (разбор подгрупп/больше данных/другие переменные/etc.)

  Ваш ответ здесь

Индивидуальная часть

На вопросы **Q27** и **Q28** давать ответы в данном документе не нужно. Отправьте их через отдельную форму (ссылка в начале документа).

Q27 (Индивидуально)

Чем занимался(лась) лично ты в команде? Понравилось ли работать в команде? Что можно было бы улучшить?

Q28 (Индивидуально)

Любые пожелания/критику/комментарии можете оставлять здесь!