Oyun Veri Analitiği Bootcamp Projesi

**Haziran 2024**

**MİNE KARA**

# Parça 1

**SORU 1: D14 LTV hesaplayınız. Yorumlayınız.**

D14 LTV değerinin hesaplanması için normalde 14 günlük bir veriye sahip olmamız gerekiyor. Ancak elimizde sadece 8 günlük veri bulunmakta. D14 LTV değeri, kullanıcının uygulamayı yükledikten sonra 14. güne kadarki süreçte getirdiği toplam gelirdir diyebiliriz.

LTV = Total cohort revenue / Cohort Size

D14 için LTV değeri ise, 14 gün boyunca elde edilen toplam gelirin, kohort sayısına bölümüdür. Tabi biz verimizden bu iki değeri net olarak bulamıyoruz. Bu durumda bir tahmin yapmamız gerekecek. Bu noktada monetization curve kullanarak 14. gün için revenue ve retention rate’i bulabiliriz.

**SORU 2: Interstitial reklam gelirinin toplam günlük gelirdeki oranı nedir? Yorumlayınız.**

Interstitial reklam gelirlerinin toplam günlük gelire oranı aşağıdaki grafikte görülebilir. Grafiğe göre, bazı günlerde interstitial reklamların toplam gelir içinde çok daha büyük bir payı olduğu görülürken, diğer günlerde bu pay daha düşük olabilmektedir. Yani farklı faktörlere bağlı (reklam kampanyaları ile ilgili olabilir) olarak bu oran değişmektedir. Ayrıca genel olarak bakıldığında, bu 8 günlük veride interstitial reklamlardan elde edilen gelirler diğer reklam türlerine (banner ve rewarded) göre daha büyük bir yüzdeyi kapsadığını görebiliriz. Bu oranlar yaklaşık %76 ile %92 arasında değişmekte.

A graph with blue lines and numbers

Description automatically generated

# 

# Parça 2

$10 CPI ile yapılan bir pazarlama kampanyasından 1000 oyuncu elde edildiğini ve bu oyuncular için Day 1 retention %50 olduğunu varsayalım (yani oyuncuların yarısı oyunu yükledikten bir gün sonra tekrar açmaktadır). Day 2 retention oranı %49. Day 3 retention %48'dir ve bu durum hiç kimse oyunu bir daha oynamayana kadar %1'lik düşüşlerle devam eder (51. Gün retention %0'dır).

**SORU 1:** **Başa baş noktasına (break even point) 50. günde ulaşılması için ortalama günlük ARPDAU (Günlük aktif kullanıcı başına ortalama gelir) ne olmalıdır? (Günler boyunca sabit olduğunu varsayalım)**

Break even point noktası LTV değerinin CPI değerine eşit olduğu gündür. Yani LTV = CPI olan günü bulmamız gerekiyor. ARPDAU değerinin de günler boyu sabit olduğu verilmiş.

Bu noktada her gün için işlem yapacak olursak eğer;

1. ARPDAU = revenue / Daily active users (DAU)
2. LTV = total cohort revenue / cohort size

Bu veride cohort size 1000 kullanıcı. Günlük revenue’yu 1. Denklemden çıkartabiliriz.

Revenue (daily) = ARPDAU (A) \* DAU

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Day** | **Retention Rate** | **Kullanıcı Sayısı** | **ARPDAU** | **Revenue (daily)** |
| D1 | 0.5 | 500 | A | 500 x A |
| D2 | 0.49 | 245 | A | 245 x A |
| D3 | 0.48 | 118 | A | 118 x A |
| D4 | 0.47 | 56 | A | 56 x A |
| D5 | 0.46 | 26 | A | 26 x A |
| D6 | 0.45 | 12 | A | 12 x A |
| D7 | 0.44 | 6 | A | 6 x A |
| D8 | 0.43 | 3 | A | 3 x A |
| D9 | 0.42 | 1 | A | 1 x A |
| D10 | 0.41 | 1 | A | 1 x A |
| **….** | **….** | **….** | **….** | **….** |
|  | **Total** | **1008** | **A** | **1008 x A** |

Kullanıcı sayıları ve günlük revenue değerleri yukarıdaki tablo görüldüğü gibi 50. Güne kadar bu şekilde ilerlemektedir. Burada total cohort revenue değeri 1008 x ARPDAU (A) olarak bulunmuştur.

LTV = CPI eşitliğine gelecek olursak

Total Cohort Revenue / Cohort Size = 10 $

1008 x A / 1000 = 10 $

A (ARPDAU) = 9.92 $ olarak bulunur.

**SORU 2: ARPDAU'nun sabit olduğunu varsaymak ne kadar gerçeği yansıtır? Eğer yansıttığını düşünüyorsanız neden? Yansıtmadığını düşünüyorsanız değişime sebep olabilecek varsayımlarınız ne olurdu?**

Ben açıkçası ARPDAU’nun sabit olarak kabul edilmesinin gerçeği tam olarak yansıtmadığını düşünüyorum. Çünkü ARPDAU değeri, tek bir faktöre bağlı olarak değişebilecek bir metrik değil. Ayrıca sadece günlük aktif kullanıcı sayısına bağlı olsaydı bile insan davranışlarının anlık değişimi bile ARPDAU’yu çok rahat etkileyebilir. Bunun haricinde kullanıcıların oyun ile etkileşimlerinde mevsimsel ya da trendsel etkiler olabilir. Mesela, haftasonu boş vakitlerinde insanlar oyunla daha çok etkileşim halinde olabilir. Ayrıca güncellemelerle değişebilecek oyun içi dinamikler de kullanıcıların oyun içi harcamalarında değişikliğe sebep olabilir.

# Parça 3

**SORU 1: Oyunda hilecileri tespit etmek istiyorsunuz bunu hangi tablodaki verileri kullanarak yaparsınız? Bu verileri kullanarak basit bir hile tespiti yapınız.**

1 - Oyuncuların bulundukları level’a göre character health ve attack değerlerini inceledim. Anormal değere sahip olan oyuncuları bulmaya çalıştım. Character health ve attack değerlerinin ortlama ve standart sapmalarını çıkarttım. Sonrasında ortalamadan 3 standart sapmadan daha uzak değerlere sahip olan oyuncuları hileli oyuncular olarak kabul ettim.

**QUERY:**

WITH stage\_stats AS (

    SELECT

        stage\_index,

        AVG(CAST(character\_health AS FLOAT64)) AS avg\_health,

        STDDEV(CAST(character\_health AS FLOAT64)) AS std\_health,

        AVG(CAST(character\_attack AS FLOAT64)) AS avg\_attack,

        STDDEV(CAST(character\_attack AS FLOAT64)) AS std\_attack

    FROM game-analysis-01.project\_game\_v2.stage\_events

    GROUP BY stage\_index

)

SELECT

    se.user\_id,

    se.stage\_index,

    CAST(se.character\_health AS FLOAT64) AS character\_health,

    CAST(se.character\_attack AS FLOAT64) AS character\_attack,

    se.event\_date

FROM game-analysis-01.project\_game\_v2.stage\_events se

JOIN stage\_stats ss ON se.stage\_index = ss.stage\_index

WHERE

    (CAST(se.character\_health AS FLOAT64) > ss.avg\_health + 3 \* ss.std\_health OR

     CAST(se.character\_health AS FLOAT64) < ss.avg\_health - 3 \* ss.std\_health) OR

    (CAST(se.character\_attack AS FLOAT64) > ss.avg\_attack + 3 \* ss.std\_attack OR

     CAST(se.character\_attack AS FLOAT64) < ss.avg\_attack - 3 \* ss.std\_attack)

ORDER BY stage\_index;

**OUTPUT: (İlk 15 satır)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| user\_id | stage\_index | character\_health | character\_attack | event\_date |
| 13357580 | 10 | 4.81E+41 | 8.29E+100 | 10/29/2023 |
| 13357580 | 10 | 4.81E+41 | 8.29E+100 | 10/29/2023 |
| 12465913 | 10 | 6.71E+96 | 2.56E+96 | 10/3/2023 |
| 12465913 | 10 | 6.71E+96 | 2.56E+96 | 10/3/2023 |
| 13357580 | 11 | 4.81E+41 | 8.29E+100 | 10/29/2023 |
| 13357580 | 11 | 4.81E+41 | 8.29E+100 | 10/29/2023 |
| 12465913 | 11 | 6.71E+96 | 2.56E+96 | 10/3/2023 |
| 12465913 | 11 | 6.71E+96 | 2.56E+96 | 10/3/2023 |
| 13357580 | 12 | 2.55E+42 | 3.81E+101 | 10/29/2023 |
| 13357580 | 12 | 2.55E+42 | 3.81E+101 | 10/29/2023 |
| 12465913 | 12 | 8.12E+96 | 2.82E+96 | 10/4/2023 |
| 12465913 | 12 | 8.12E+96 | 2.82E+96 | 10/4/2023 |
| 13256277 | 13 | 4.56E+31 | 1.30E+30 | 10/25/2023 |
| 13256277 | 13 | 4.56E+31 | 1.30E+30 | 10/25/2023 |

Tabloda görüldüğü üzere level 11’de olan bu oyuncuların bu 4.81x1041 character health’e sahip olması mantıklı gelmiyor. Aynı mantık diğer oyuncular için de geçerlidir.

2 - Farklı bir analiz daha gerçekleştirdim. Bu sefer, oyuncuların bulundukları level’a göre currency change amount’u inceledim. Yine aynı mantıkta ortlamadan uzak noktaları bularak anaormal davranış gösteren oyuncuların çıktısını aldım.

**QUERY:**

WITH currency\_stats AS (

    SELECT

        se.stage\_index,

        AVG(CAST(cc.currency\_change\_amount AS INT64)) AS avg\_currency,

        STDDEV(CAST(cc.currency\_change\_amount AS INT64)) AS std\_currency

    FROM game-analysis-01.project\_game\_v2.stage\_events se

    JOIN game-analysis-01.project\_game\_v2.currency\_changes cc

    ON se.user\_id = CAST(cc.user\_id AS INT64)

    GROUP BY se.stage\_index

)

SELECT

    cc.user\_id,

    se.stage\_index,

    cc.currency\_change\_amount,

    cc.event\_date

FROM game-analysis-01.project\_game\_v2.currency\_changes cc

JOIN game-analysis-01.project\_game\_v2.stage\_events se

ON CAST(cc.user\_id AS INT64) = se.user\_id

JOIN currency\_stats cs

ON se.stage\_index = cs.stage\_index

WHERE

    CAST(cc.currency\_change\_amount AS INT64) > cs.avg\_currency + 3 \* cs.std\_currency OR

    CAST(cc.currency\_change\_amount AS INT64) < cs.avg\_currency - 3 \* cs.std\_currency

ORDER BY stage\_index;

**OUTPUT: (İlk 15 satır)**

|  |  |  |  |
| --- | --- | --- | --- |
| user\_id | stage\_index | currency\_change\_amount | event\_date |
| 12927055 | 1 | 87699872 | 10/19/2023 |
| 12927055 | 1 | 87699872 | 10/19/2023 |
| 12927055 | 1 | 87699872 | 10/19/2023 |
| 12927055 | 1 | 87699872 | 10/19/2023 |
| 12927055 | 1 | 87699872 | 10/19/2023 |
| 12927055 | 1 | 87699872 | 10/19/2023 |
| 12927055 | 1 | 87699872 | 10/19/2023 |
| 12927055 | 1 | 87699872 | 10/19/2023 |
| 12927055 | 1 | 87699872 | 10/19/2023 |
| 12927055 | 1 | 87699872 | 10/19/2023 |
| 12927055 | 1 | 87699872 | 10/19/2023 |
| 12927055 | 1 | 87699872 | 10/19/2023 |
| 12927055 | 1 | 87699872 | 10/19/2023 |
| 12927055 | 1 | 87699872 | 10/19/2023 |

A graph with blue squares

Description automatically generated

Yukarıdaki grafikte görüldüğü üzere 100. leveldakiler (ki onlar için bile bence büyük bir currency değişimi) currency değişiminin çok olduğu levellar genelde ilk 10 level. Bu level’larda olan kullanıcıların currency değişimlerinin bu kadar fazla olamayacağı barizdir. Bu yüzden bu oyuncular da hileli oyuncular kategorisine alınır.

**SORU 2: Mevcut verilere ve bilgilere göre sizce oyuncuların gem kazanabileceği en iyi kaynak hangisidir?**

En çok gem kazandıran kaynakları “currency\_changes” tablosundaki reason, currency change amount, currency type ve change type verilerine bakarak inceleyebiliriz.

**QUERY:**

SELECT

    reason,

    COUNT(\*) AS num\_transactions,

    AVG(CAST(cc.currency\_change\_amount AS INT64)) AS avg\_gems\_per\_transaction,

    SUM(CAST(cc.currency\_change\_amount AS INT64)) AS total\_gems

FROM game-analysis-01.project\_game\_v2.currency\_changes cc

WHERE currency\_type = 'Gem' AND change\_type = 'Gain'

GROUP BY reason

ORDER BY total\_gems DESC;

**OUTPUT:**

|  |  |  |  |
| --- | --- | --- | --- |
| **reason** | **num\_transactions** | **avg\_gems\_per\_transaction** | **total\_gems** |
| login | 36811 | 228267.1913 | 8402743580 |
| quest\_new\_user | 4885 | 839466.8393 | 4100795510 |
| market\_offer | 5703 | 526115.5935 | 3000437230 |
| other\_01 | 15541 | 135226.4333 | 2101554000 |
| quest | 532374 | 3763.420941 | 2003547460 |
| enemy\_kill\_02 | 25824 | 77273.71577 | 1995516436 |
| game\_mode\_01 | 22838 | 44327.36229 | 1012348300 |
| chest\_05 | 93991 | 4484.956006 | 421545500 |
| chest\_02 | 129758 | 2337.281709 | 303281000 |

A graph with red bars

Description automatically generated

A graph with red bars

Description automatically generated

Yukarıdaki grafiklere göre;

En yüksek toplam gem kazancını “login” işlemi sağlamaktadır. Yani düzenli giriş yapmak oyuncular için önemli bir gem kaynağıdır diyebiliriz.

Transaction başına ortalama en yüksek gem kazancı sağlayan kaynak ise, “quest\_new\_user” işlemidir. Oyunda yeni kullanıcılara önemli ödüller sağlandığı görülmektedir.

“quest” işleminde ise total gem sayısı fazla olsa da ortalamaya bakıldığında düşük kazanç sağladığı görülür. Yani quest’ler yapılıyor ancak çok büyük bir gem getirisi sağlamıyor.

**SORU 3: Day5’te hala oyunda olan bir oyuncunun hangi stage’de olması en muhtemeldir?**

**QUERY:**WITH day5\_active\_users AS (

    SELECT CAST(ud.user\_id AS INT64) AS user\_id

    FROM game-analysis-01.project\_game\_v2.users\_daily ud

    JOIN game-analysis-01.project\_game\_v2.users u

    ON CAST(ud.user\_id AS INT64)= u.user\_id

    WHERE ud.event\_date = DATE\_ADD(u.install\_date, INTERVAL 5 DAY)

),

stage\_counts AS (

    SELECT

        se.user\_id,

        se.stage\_index,

        COUNT(\*) AS stage\_count

    FROM game-analysis-01.project\_game\_v2.stage\_events se

    JOIN day5\_active\_users dau ON se.user\_id = dau.user\_id

    GROUP BY se.user\_id, se.stage\_index

)

SELECT

    stage\_index,

    COUNT(\*) AS num\_users

FROM stage\_counts

GROUP BY stage\_index

ORDER BY num\_users DESC

LIMIT 5;

A graph with a bar

Description automatically generated with medium confidence

Yukarıdaki grafiğe göre Day5’te hala oyunda olan bir oyuncunun en muhtemel stage’i 1’dir. Yani oyuncuların en sık bulunduğu aşama stage 1’dir. Stage index numarası yanlış anlamadıysam oyuncunun bulunduğu bölümü belirtiyor. Bu noktada, gün 5’te oyuncuların hala bölüm 1’de olması beni şaşırttı. Eğer oyunda uzun bir tutorial süreci varsa belki de oyuncular ilk birkaç gün tutorial’ı oynamaklar geçiriyor olabilir.

**SORU 4: Kümülatif olarak 15. denemesini (attempt) gerçekleştiren bir oyuncu kaçıncı bölümde (stage) olması muhtemeldir?**

**QUERY:**

WITH stage\_attempts AS (

    SELECT

        user\_id,

        stage\_index,

        SUM(CAST(stage\_attempt AS INT64)) AS total\_attempts

    FROM game-analysis-01.project\_game\_v2.stage\_events

    GROUP BY user\_id, stage\_index

),

cumulative\_att AS (

    SELECT

        user\_id,

        stage\_index,

        total\_attempts,

        SUM(total\_attempts) OVER (PARTITION BY user\_id ORDER BY stage\_index) AS cumulative\_attempts

    FROM stage\_attempts

)

SELECT

    stage\_index,

    COUNT(\*) AS num\_users

FROM cumulative\_att

WHERE cumulative\_attempts >= 15

GROUP BY stage\_index

ORDER BY num\_users DESC

LIMIT 1;

**OUTPUT:**

|  |  |
| --- | --- |
| stage\_index | num\_users |
| 1 | 47028 |

15. denemesini gerçekleştiren oyuncuların en muhtemel olduğu aşama stage\_index = 1’dir.

**SORU 5: Veri setini inceleyerek kendi analiz etmek istediğiniz bir soruyu oluşturarak cevabını ve yorumlarınızı paylaşınız.**

**Soru: Oyunda kalma süresi ile gem ve gold ilişkisini inceleyeceğim.**

**QUERY:**

WITH player\_retention AS (

    SELECT

        u.user\_id,

        MIN(ud.event\_date) AS first\_play\_date,

        MAX(ud.event\_date) AS last\_play\_date,

        DATE\_DIFF(MAX(ud.event\_date), MIN(ud.event\_date), DAY) AS days\_active

    FROM game-analysis-01.project\_game\_v2.users u

    JOIN game-analysis-01.project\_game\_v2.users\_daily ud

    ON u.user\_id = CAST(ud.user\_id AS INT64)

    GROUP BY u.user\_id

),

gem\_gold AS (

    SELECT

        pr.user\_id,

        pr.days\_active,

        us.current\_gem,

        us.current\_gold,

    FROM player\_retention pr

    JOIN game-analysis-01.project\_game\_v2.user\_states us

    ON pr.user\_id = us.user\_id

    WHERE us.event\_date = pr.last\_play\_date

)

SELECT

    gg.days\_active,

    AVG(SAFE\_CAST(gg.current\_gem AS FLOAT64)) as avg\_gem,

    AVG(SAFE\_CAST(gg.current\_gold AS FLOAT64)) as avg\_gold,

FROM gem\_gold gg

JOIN game-analysis-01.project\_game\_v2.user\_states us

ON gg.user\_id = us.user\_id

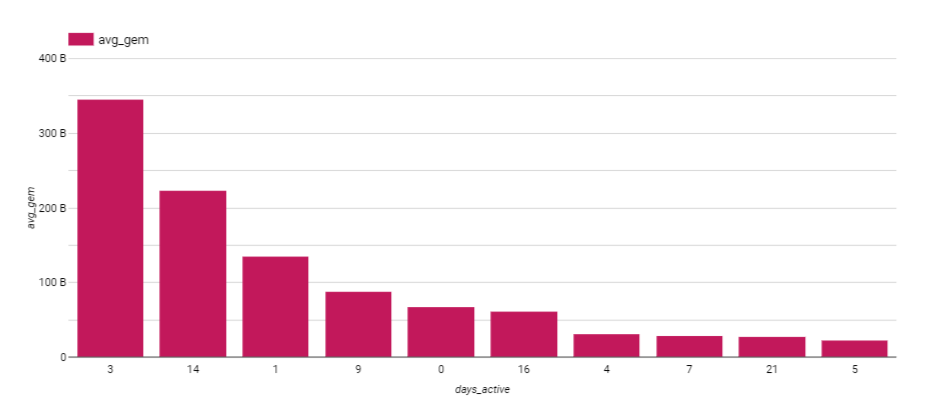
GROUP BY gg.days\_active

ORDER BY gg.days\_active DESC;

OUTPUT:

|  |  |  |
| --- | --- | --- |
| days\_active | avg\_gem | avg\_gold |
| 3 | 345794.8998 | 180632.9947 |
| 14 | 223560.9891 | 29193.5753 |
| 1 | 134972.2953 | 123154.4956 |
| 9 | 88025.86874 | 54976.62768 |
| 0 | 68159.86226 | -60702.34941 |
| 16 | 61311.37464 | 8086.483947 |
| 4 | 31761.97471 | 35617.57268 |
| 7 | 29425.78517 | 48220.59729 |
| 21 | 27541.31225 | 23760.55467 |

**Days active - Average Gem grafiği:**



İlk üç gün içinde oyunda kalan oyuncuların ortalama gem miktarları oldukça yüksek görünüyor (345,794.8998). Bu, yeni oyuncuların başlangıçta daha fazla gem elde ettiğini veya bu oyuncuların oyun içi satın alım yaptığını gösterebilir.

14 gün oyunda kalan oyuncuların ortalama gem miktarı da ilk üç günkü kadar yüksek olmasa da iyi bir seviye de ancak arada 16 gün oyunda kalan oyuncuların gem miktarları nispeten düşük. 14. gün oyuna gelen oyunculara önümüzdeki 2 gün içerisinde oyundaki gem artırmaya yönelik stratejiler geliştirilebilir.

**Days active - Average Gold grafiği:**

A graph with lines and numbers

Description automatically generated

41 gün aktif olan oyuncuların yüksek miktarda gold kazandığı görülüyor. Bunun sebebi, o gün oyunculara özel etkinlikler veya promosyonlar sunulmuş olabilir. Ya da oyun geliştiricileri tarafından 41. güne ulaşan kişilere ekstra gold dağıtımı yapılmış olabilir.