



UNIVERSIDADE FEDERAL DE PERNAMBUCO

Centro de Informática - Graduação em Ciência
da Computação

Análise do uso de ajuste de curvas e de outras
métricas na criação de curvas de tendências
para jogos baseadas na sua renda bruta

Trabalho de Conclusão de Curso - 2017.2

Aluno: Túlio Paulo Lages da Silva (tpls@cin.ufpe.br)

Orientador: Geber Lisboa Ramalho (glr@cin.ufpe.br)

Recife, 2017

Agradecimentos

Agradeço primeiramente aos meus pais, que me dão todo o suporte necessário para seguir em frente. Agradeço também aos meus gatos e a minha cachorra por todas as alegrias e estresses diários que me fazem sentir vivo.

Agradeço a Geber por ser muito gentil e empático, me auxiliando em momentos decisivos deste trabalho e, quando pedido, sempre me dando suporte, até quando eu talvez não merecesse.

Agradeço a todos os meus amigos do trabalho e da faculdade (em especial Sarah, Mariama, Fanny, Gabriela, Déborah, Daniel, Leandro, Matheus, Folha, Igor, Ferrão, Peão, Bruna, Vinícius, Victor, André, Carneiro, Tomás, Nicolle, José Luiz, Luiz, Aline, Gisely, Bessa, Miguel, Robertson, Duda, Cisneiros, Joselito e Priscila), que me ajudaram a suportar toda a dor e desafios e por compartilhar os bons e maus momentos.

Agradeço a Leando e a Daniel por serem companheiros em todos os momentos e por todas as experiências compartilhadas nestes anos, assim como o suporte nas etapas finais.

Agradeço a Sarah e a tia Magna por me darem força para terminar este trabalho e por serem pessoas maravilhosas.

Agradeço a Fanny por ser uma inspiração e uma pessoa incrível, me ajudando a encarar os desafios e dificuldades do dia a dia com a cabeça erguida.

Agradeço a Victor e a Vinícius por serem amigos que sempre estão presentes, cuidando da minha saúde mental.

Por último, agradeço a todos os professores que já me deram aula na UFPE. De alguma forma, cada um contribuiu para o que sou hoje e, por isso, sou eternamente grato.

Sumário

1. Introdução	9
1.1 Motivação	9
1.2. Objetivos	10
1.3. Organização do trabalho	11
2. Referencial teórico	12
2.1. MAU/DAU	12
2.2. Modelo de negócio	12
2.3. Premium/F2P	12
2.4. SaaS	13
2.5. Otimização de Taxa de Conversão	13
2.6. App Store Optimizator	13
2.7. Algoritmo evolucionário e algoritmo genético	14
2.8. Ajuste de curva	15
2.9. Modelo Cliente-Servidor	15
2.10. Sqlite/Sequelize/ORM	16
2.11. Angular	17
2.12. Nodejs	17
3. Descrição dos problemas	17
3.1. Previsão de viabilidade de jogos	17
3.2. Acessibilidade a Business Intelligence	20
4. Estado da arte	21
4.1. AppAnnie	21
4.2. SensorTower	22
4.3. Comparativo	23
5. Proposta	24
5.1 Curvas de tendência sobre renda bruta	24
5.2. Método e fonte de obtenção dos dados	25
6. Implementação	26
6.1 Arquitetura	26
6.1.1 . Servidor	27
6.1.2 . Cliente web	28
6.2. Resultados	29
7. Conclusões	31
8. Referências	32

Resumo

O mercado de jogos móveis está cada vez mais abrangente e competitivo, o que torna cada vez mais importante a obtenção de informações relevantes sobre o ciclo de vida de jogos durante as tomadas de decisões de projeto. Este trabalho visa explorar a criação de uma ferramenta de criação de linhas e curvas de tendência usando o histórico de posições (classificação por receita bruta) de jogos da Play Store, estudando se essa métrica é relevante e que outras métricas úteis poderiam surgir a partir desse tipo de dado.

Palavras chaves:

ASO, playstore, mobile, games, trending-curve, curve-fitting

Lista de figuras

Identificador	Descrição	Página
Figura 1	Máximo local vs máximo global	12
Figura 2	Ciclo de vida de um algoritmo genético	13
Figura 3	Arquitetura cliente-servidor	14
Figura 4	Instalações mensais por gênero de jogo móvel	16
Figura 5	MAUs por gênero de jogo móvel	17
Figura 6	Jogos móveis por renda bruta total	18
Figura 7	Dashboard de analytics de um dado aplicativo no App Annie	20
Figura 8	Dashboard de ASO do SensorTower	21
Figura 9	Exemplo de json obtido através API da PlayStore	24
Figura 10	Arquitetura completa do sistema	24
Figura 11	Página inicial do sistema web.	26
Figura 12	Exemplo de linha computada a partir de um algoritmo genético.	27
Figura 13	Tela de customização da criação das curvas de tendência.	27
Figura 14	Linha gerada pelo sistema para um jogo selecionado.	28
Figura 15	Curva gerada pelo sistema para um jogo selecionado.	28
Figura 16	Gráficos do jogo de teste de resultados	29
Figura 17	Gráficos dos dois jogos de testes de resultado	30

Lista de tabelas

Identificador	Descrição	Página
Tabela 1	Comparativo entre os ASOs App Annie e Sensor Tower	21
Tabela 2	Lista de endpoints do sistema proposto	25

1. Introdução

Neste capítulo, será dada uma breve introdução sobre o trabalho proposto, como sua motivação, como está estruturado e os objetivos esperados para facilitar o entendimento do leitor.

1.1 Motivação

Com a popularização de dispositivos móveis e os constantes avanços tecnológicos voltados a diminuir cada vez mais a diferença de desempenho entre computadores domésticos e smartphones, o mercado de jogos móveis está cada vez mais abrangente. Além disso, o desenvolvimento de jogos móveis está cada vez mais acessível, o que gera atualmente uma explosão de jogos nas lojas de aplicativos de todas as plataformas. Com isso, a concorrência entre os milhares de jogos móveis aumenta drasticamente, criando uma necessidade cada vez maior de se entender quais fatores são responsáveis pelo sucesso e fracasso desses aplicativos, assim como entender o ciclo de vida dos mesmos.[1]

Há, no entanto, pontos a serem explorados e melhorados nas ferramentas atuais que possibilitem um estudo mais detalhado do estado do mercado e desses fatores que influenciam as variações na popularidade de jogos, o que é necessário tanto para quem quer entender e gerenciar o ciclo de vida de seu jogo assim como para quem quer entender, por exemplo, que tipo de jogo valeria mais a pena ser desenvolvido a médio e longo prazo. Existem algumas plataformas que são usadas para auxiliar na análise e descoberta de dados sobre aplicativos móveis, incluindo jogos, como os produtos do App Annie, SensorTower e 42matters, assim como alguns estudos que tentam desvendar relações implícitas entre dois ou mais jogos e nível de popularidade ou lucro bruto, mas ainda há abordagens e métricas possivelmente úteis a serem melhor estudadas.

1.2. Objetivos

Este trabalho busca estudar e entender melhor as necessidades do mercado atual de jogos para dispositivos móveis e propor novas formas de visualizar dados disponíveis deste mercado, visando complementar ferramentas que já existem. Para isto, o foco maior será tentativa de analisar as tendências para determinados jogos e suas categorias, inicialmente testando-se que tipo de informações úteis são proporcionadas a partir da geração de linhas e curvas de tendência criadas a partir do ranking por receita bruta diário dos jogos nas lojas de aplicativos. A partir da curva gerada, pretende-se estudar se o jogo ou categoria em questão está em declínio, em ascensão ou estável, e possivelmente derivar novas informações a partir disso. No fim deste trabalho, espera-se ter um projeto de código aberto em JavaScript que sirva como base para a obtenção de dados de jogos móveis e para a criação de um ASO e que auxilie na criação, customização e visualização de gráficos contendo todo o tipo de informação testada neste estudo, visando criar mais uma alternativa no estudo do ciclo de vida de jogos móveis, suas categorias e subgrupos.

1.3. Organização do trabalho

Este trabalho está dividido em 7 capítulos, descritos a seguir.

O capítulo 1 consiste de uma breve introdução para contextualizar o leitor, explicando a motivação e os objetivos por trás do trabalho.

O capítulo 2 consiste do referencial teórico para que o leitor entenda os conceitos utilizados na explicação do problema, do estado da arte e da solução proposta.

O capítulo 3 expande a breve introdução ao problema do capítulo 1, explicando melhor a situação atual do mercado de jogos móveis e as dificuldades e necessidades que os desenvolvedores do setor enfrentam.

O capítulo 4 explora um pouco as atuais ferramentas utilizadas para análise e otimização do mercado de jogos móveis, assim como suas vantagens e desvantagens.

O capítulo 5 consiste da proposta do trabalho, explicando quais problemas ela ataca e como ela os resolve.

O capítulo 6 consiste de detalhes sobre a implementação da ferramenta de criação de curvas de tendências, elaborando sobre os algoritmos, escolhas arquiteturais e detalhes do código desenvolvido.

O capítulo 7 consiste de uma breve conclusão, assim como trabalhos futuros, seguido das referências.

2. Referencial teórico

Neste capítulo serão explicados alguns termos e conceitos importantes para melhor entendimento do leitor em relação ao trabalho. Como há uma multidisciplinaridade inerente do tema, os conceitos serão divididos entre três partes principais. Primeiro, serão tratados conceitos relacionados ao mercado de jogos e aplicativos móveis. Em seguida, serão explicados termos relacionados à solução proposta, tanto na parte técnica como na parte teórica.

2.1. MAU/DAU

Usuários Ativos Diários (“**Daily Active Users**” em inglês, ou **DAU**) é uma métrica que calcula a quantidade de usuários únicos na janela de um dia que utilizam um dado app.

Usuários Ativos Mensais (“**Monthly Active Users**”, ou **MAU**) é uma métrica que conta o total de usuários únicos ativos no período de um único mês (normalmente considera-se que o mês possui 30 dias).

A razão **MAU/DAU** é uma referência importante para a indústria, servindo de métrica padrão para retenção de usuários. Por exemplo, uma razão de 1 significa que todo usuário que utilizou um dado app no mês passado está ativo novamente hoje. Para apps móveis, 20% DAU/MAU é geralmente considerado um bom número[19].

2.2. Modelo de negócio

Um modelo de negócio é a parte de uma estratégia de negócio que define a lógica de criação, entrega e captura de valor por parte de um grupo ou organização em contextos econômicos, culturais ou sociais.[19]

2.3. Premium/F2P

Premium é um modelo de negócio no qual o consumidor final instala e utiliza um aplicativo somente após pagar por ele.

Freemium ou **Free To Play (F2P)** é um modelo de negócio no qual o consumidor final instala e joga o “*core loop*” de um jogo gratuitamente mas precisa pagar por bens e moedas virtuais através de microtransações.[21]

2.4. SaaS

Software como um Serviço (**Software as a Service** ou **SaaS** em inglês)[22] é uma forma de entregar aplicações centralmente hospedadas pela Internet, só que como um serviço. Aplicações SaaS normalmente são chamadas de software baseados na web, software sob demanda ou até mesmo software hospedado. As aplicações SaaS rodam nos servidores dos provedores da aplicação. Em vez de se instalar e manter um software, o usuário passa a acessá-lo pela Internet, tirando a sua necessidade de gerenciar software e hardware. As responsabilidades de segurança, disponibilidade e performance ficam sob cuidado do provedor do serviço. Normalmente o acesso a este tipo de serviço se dá através de um cliente web.

2.5. Otimização de Taxa de Conversão

Otimização de Taxa de Conversão (do inglês “Conversion Rate Optimization”, ou CRO) é uma forma estruturada e sistemática de melhorar a performance de um site, e extrair mais do tráfego que você já tem. Ou seja, ter mais conversões sem necessariamente precisar aumentar o tráfego.[20]

2.6. App Store Optimizator

App store optimizer (ASO) [14, 15] é o processo de melhorar a visibilidade de um app móvel em uma dada loja de apps (como Google Play para Android e iTunes para iOS), de forma similar com o que um Search engine optimization (SEO) faz para sites web. Para este fim específico, ASOs incluem o processo de tentar fazer com que o app mantenha uma posição alta nos resultados da busca na loja e nos gráficos de top rankings.

ASOs são compreendidos normalmente, no mínimo, por dois componentes: dados históricos das aplicações de uma dada loja e formas diferentes de visualizá-las. Sua principal funcionalidade é a de prover informação contextual relevante para criadores de aplicações, sejam jogos ou não. Além disto, ASOs englobam atividades focadas em aumentar a conversão de impressões de apps na loja em downloads (ou seja, diminuir o custo de conversão de impressões em usuários), o que é chamado de **otimização de taxa de conversão**. É consenso entre companhias de marketing móvel que posições mais altas nos resultados de busca resultam em mais downloads para um dado app.

Ou seja, o trabalho de um ASO resume-se, no fim das contas, a otimizar as palavras-chave de um dado app para que sua posição no ranking aumente em uma busca (uma vez que por volta de 65% de todos os downloads se dão a partir de resultados de busca nas lojas) com para palavras-chave relacionadas e aumentar o CRO do app.

2.7. Algoritmo evolucionário e algoritmo genético

Algoritmos evolucionários usam modelos computacionais dos processos naturais de evolução como meio de resolução de problemas. Mesmo com a vasta gama de modelos propostos, todos têm uma base em comum: o conceito de simulação da evolução das espécies através da seleção, mutação e reprodução, que dependem do desempenho do candidato num dado ambiente[18].

Algoritmos genéticos (GA, ou “Genetic Algorithms” em inglês) são um ramo dos algoritmos evolucionários e como tal podem ser definidos como uma técnica de busca baseada numa metáfora do processo biológico de evolução natural[18].

Os algoritmos genéticos são técnicas heurísticas de otimização global. A questão da otimização global opõe os GAs aos métodos como gradiente(hill climbing), que seguem a derivada de uma função de forma a encontrar o máximo de uma função, ficando facilmente retidos em máximos locais[Figura 1].

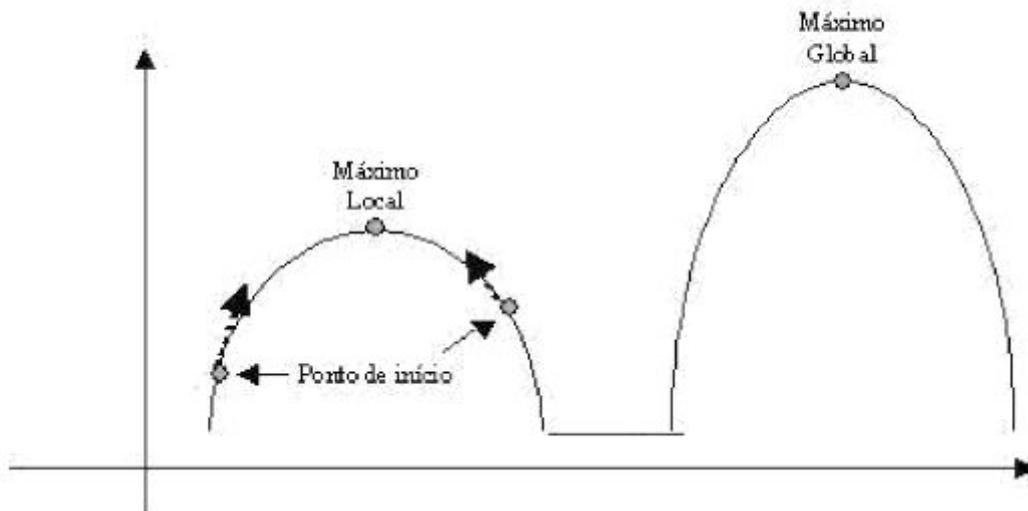


Figura 1: Função hipotética com um máximo local e outro global. Uma técnica de *hill climbing* se inicia em qualquer um dos pontos de início indicados seguirá o gradiente (direção de maior crescimento) e acabará no ponto de máximo local (onde a derivada é zero). Algoritmos genéticos não têm essa dependência tão forte dos valores iniciais, portanto estão menos propensos (porém não imunes) a cair em máximos locais.

Nos algoritmos genéticos, populações de indivíduos são criados e submetidos aos operadores genéticos de seleção, recombinação (crossover) e mutação, que utilizam uma caracterização de qualidades de cada indivíduo como solução (avaliação) do problema em questão e geram um processo de evolução natural destes indivíduos, que eventualmente deverá gerar um indivíduo que caracterizará uma boa (ou a melhor) solução para o dado problema[Figura 2].

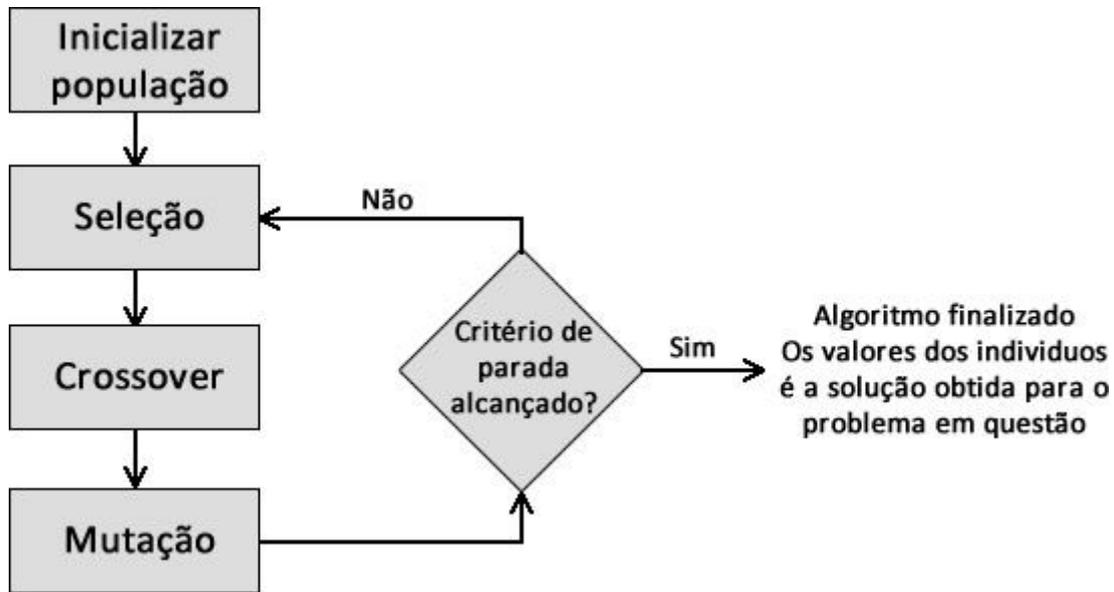


Figura 2: Ciclo de vida de um algoritmo genético

Ou seja, algoritmos genéticos são algoritmos de busca baseados nos mecanismos de seleção natural e genética, combinando fatores de sobrevivência entre os melhores candidatos com uma forma estruturada de troca de características/informação genética entre dois indivíduos selecionados para formar uma estrutura heurística de busca.

2.8. Ajuste de curva

Ajuste de Curvas é um método que consiste em encontrar uma curva que se ajuste a uma série de pontos e que possivelmente cumpra uma série de parâmetros adicionais. Ajuste de curvas pode envolver tanto interpolação, onde é necessário um ajuste exato aos dados, quanto suavização, na qual é construída uma função "suave" que se aproximadamente se ajusta aos dados. Outro assunto relacionado é análise de regressão, a qual se foca mais em questões da inferência estatística[16].

O ajuste de curvas é muito utilizado para, a partir de dados conhecidos, fazer-se extrapolações. Por exemplo, conhece-se os dados de consumo anual de carga elétrica de uma cidade. A partir destes dados conhecidos, pode-se fazer projeções para o futuro e com isso, fazer-se um planejamento para que a cidade seja suprida de forma adequada nos anos subsequentes. A ideia é ajustar uma curva que melhor se ajusta aos dados disponíveis. Conhecida a equação da curva, pode-se determinar valores fora do intervalo conhecido.

2.9. Modelo Cliente-Servidor

Um modelo cliente-servidor é uma estrutura de aplicações distribuídas que particiona as tarefas e cargas de trabalho entre provedores de um recurso (servidores) e os consumidores deste recurso (clientes). Geralmente clientes e servidores comunicam-se através de uma rede de computadores, uma vez que podem estar em hardwares separados.[23]

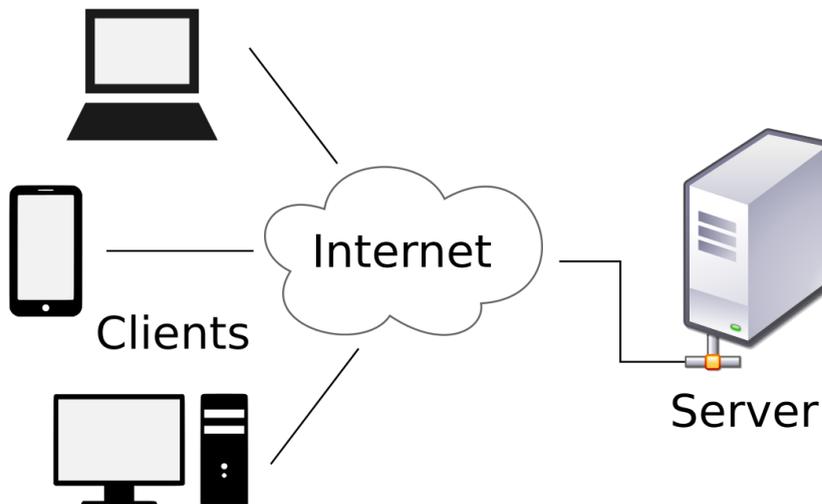


Figura 3: Arquitetura cliente-servidor: o cliente comunica-se com o servidor através da Internet, uma vez que, normalmente, cada um está em uma máquina separada, para obter dados da aplicação sendo executada.

2.10. Sqlite/Sequelize/ORM

Mapeamento objeto-relacional (ou ORM, do inglês: Object-relational mapping) é uma técnica de desenvolvimento utilizada para reduzir a impedência da programação orientada aos objetos utilizando bancos de dados relacionais. As tabelas do banco de dados são representadas através de classes e os registros de cada tabela são representados como instâncias das classes correspondentes.

Sequelize é um ORM baseado em promessas para Node.js. Ele dá suporte a vários dialetos, como PostgreSQL, MySQL, SQLite e MSSQL, além de ter suporte a transações sólidas, relações, replicação de leitura e mais.

SQLite é uma biblioteca que implementa um motor de banco de dados SQL autocontido, sem servidor, sem configuração necessária e transacional. O código para SQLite está no domínio público e é, portanto, gratuito para qualquer tipo de uso, comercial ou privado. SQLite é o banco de dados mais amplamente publicado no mundo em mais aplicações do que se pode contar, incluindo vários projetos sofisticados e importantes. Por ser um motor de banco de dados SQL embarcado, ele não tem um processo diferente no servidor, diferentemente dos outros banco de dados. SQLite lê e escreve diretamente no disco de dados. O formato de arquivo da base de dados é *cross-platform*, ou seja, uma base de dados pode ser livremente copiada e utilizada entre sistemas de 32 e 64 bits ou entre arquiteturas big-endian e little-endian. [25]

2.11. Angular

Angular (conhecido como "**Angular 2+**" ou "**Angular 2**") é uma plataforma de front-end de aplicações web de código aberto baseada em TypeScript desenvolvida pela Equipe Angular da Google e por uma comunidade de indivíduos e corporações.

2.12. Nodejs

Node.js é um interpretador de código JavaScript que funciona do lado do servidor. Seu objetivo é ajudar programadores na criação de aplicações de alta escalabilidade (como um servidor web), com códigos capazes de manipular dezenas de milhares de conexões simultâneas, numa única máquina física. O Node.js é baseado no interpretador V8 JavaScript Engine (interpretador de JavaScript open source implementado pelo Google em C++ e utilizado pelo Chrome).

3. Descrição dos problemas

Neste capítulo, será dada uma visão geral do estado atual do mercado de jogos móveis e os principais desafios enfrentados atualmente pelas pessoas deste setor.

3.1. Previsão de viabilidade de jogos

Enquanto o mercado de apps móveis está mais competitivo do que nunca, as oportunidades continuam atrativas. Por exemplo, donos de smartphones instalam jogos móveis mais do que qualquer outro tipo de aplicativo durante a sua primeira semana após a obtenção do aparelho. Para se conseguir uma vantagem neste mercado saturado, publishers precisam de algum tipo de inteligência competitiva que revele as oportunidades mais atrativas, assim como as de maior risco. Estas estatísticas também fornecem referências que auxiliam desenvolvedores a medir o sucesso de seus aplicativos e reconhecer áreas onde se é possível melhorar.[5]

Atualmente, há várias métricas que auxiliam neste fim. Com dados para todo tipo de jogo móvel, torna-se possível identificar as melhores e piores oportunidades no ecossistema de jogos móveis e estabelecer certos padrões de sucesso e fracasso.

Quando se discute sobre a saúde do negócio, publishers de jogos móveis geralmente citam estatísticas de download. Afinal de contas, todas as outras métricas e comportamentos de jogos móveis dependem dos downloads, uma vez que, sem eles, nenhum outro dado pode ser coletado.

Como demonstrado na figura 1, em julho de 2016, o gênero de Ação e Arcade tiveram mais que o dobro de downloads que todas as demais categorias unidas. Enquanto este tipo de dado abre as portas para várias análises, normalmente não se consegue chegar a nenhuma conclusão definitiva do porquê um dado gênero consegue mais instalações que um outro. Há muitos fatores que influenciam, de forma variada e ainda, por vezes, obscura, como jogos e categorias de jogos se comportam. Seja por terem sido projetados com tal propósito, seja por acidente, fatores como aquisição paga, boca a boca, estar destacado das lojas de aplicativos, parcerias e campanhas de marketing podem impactar ou não nas estatísticas de download.[5]

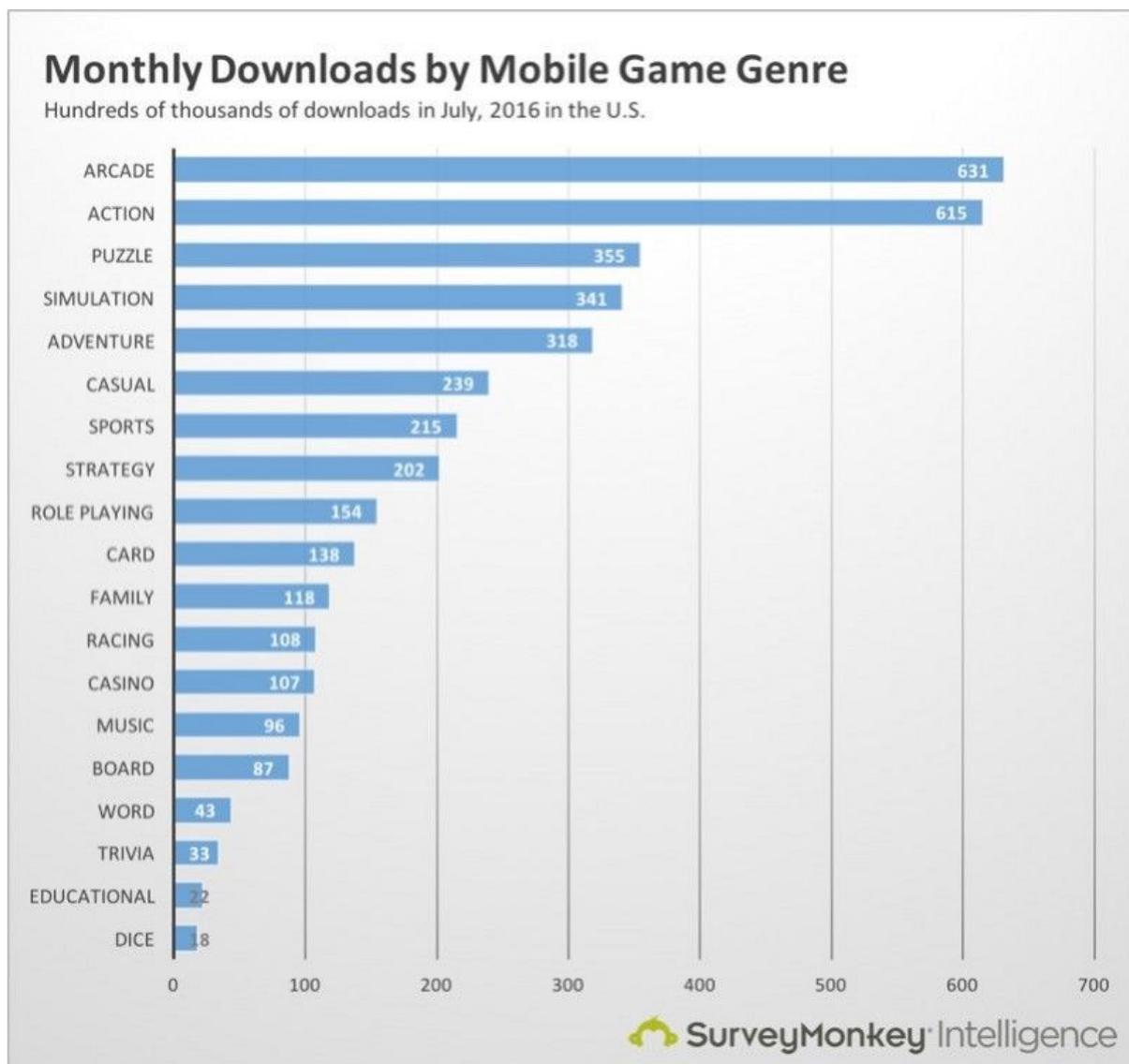


Figura 4: Instalações mensais por gênero de jogo móvel.

Além disto, apesar de conseguir várias instalações para um dado jogo ser uma boa façanha, é necessário mais que isso para se fazer dinheiro. É preciso engajar os usuários para que se tenha uma boa base de usuários ativos pagantes em todos os momentos.

Como mostra a figura 2, se for feita uma correlação entre a quantidade de usuários ativos e os downloads, apesar do gênero Arcade estar no topo em ambos os gráficos, pouca relação

existe entre os dados apresentados. Ou seja, conseguir vários downloads não garante que muitos usuários ativos serão obtidos. Os jogadores mais ávidos gastam dezenas de horas em seus smartphones, mas o tempo médio que o jogador casual gasta por mês é muito menor que isto, chegando a menos de uma hora por mês para muitos apps. Isto pode prover alguns insights para desenvolvedores, como criar mais conteúdo e um loop central de interações que seja divertido e atrativo para os jogadores, deixando de focar apenas na obtenção de usuários, mas também em sua manutenção.[5]

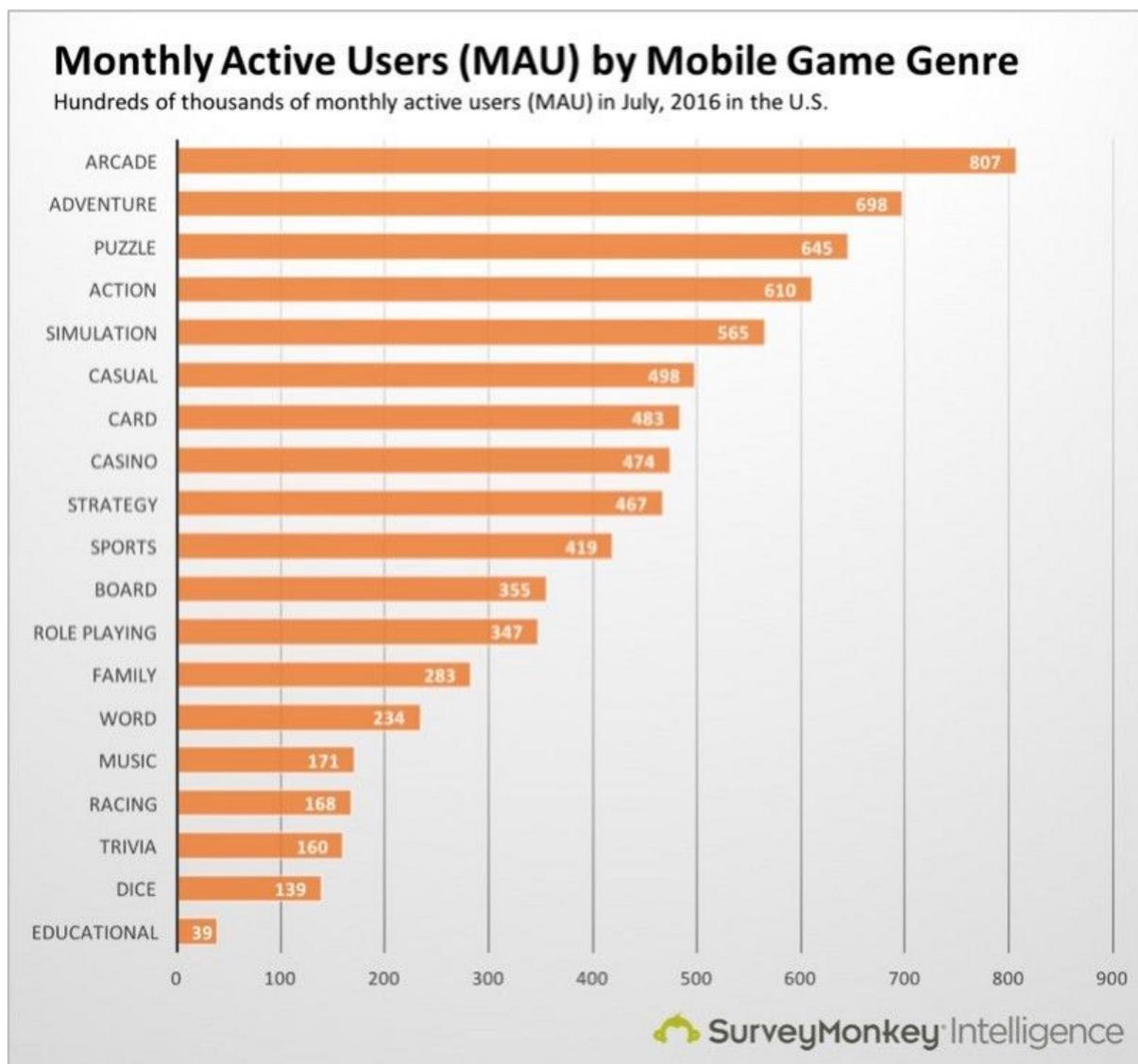


Figura 5: MAUs por gênero de jogo móvel.

[2] Demonstra, no entanto, que apesar de quantidade de instalações ser uma boa métrica para se estudar a relação entre performance e características de um dado jogo móvel, provavelmente devido aos fatores citados acima, em conjunto com os grandes gastos feitos durante o lançamento dos jogos, a renda bruta total traz algumas relações positivas. Ela demonstra ser relacionável com *hard currency gambling*, *time skips*, *request friend help*, *soft currency*, *leaderboard*, *facebook*, *consumables and levels*. Sendo uma métrica teoricamente com um relacionamento mais claro com características de jogos, ela se torna mais desejável na criação de gráficos para estudo do mercado.

Enquanto o mercado móvel inclui a receita de jogos de tablets, o setor é claramente dominado por smartphones. Dos \$46 bilhões esperados para o ano de 2017, estima-se que \$35.3 bilhões virão de jogos de smartphones[1, 3].

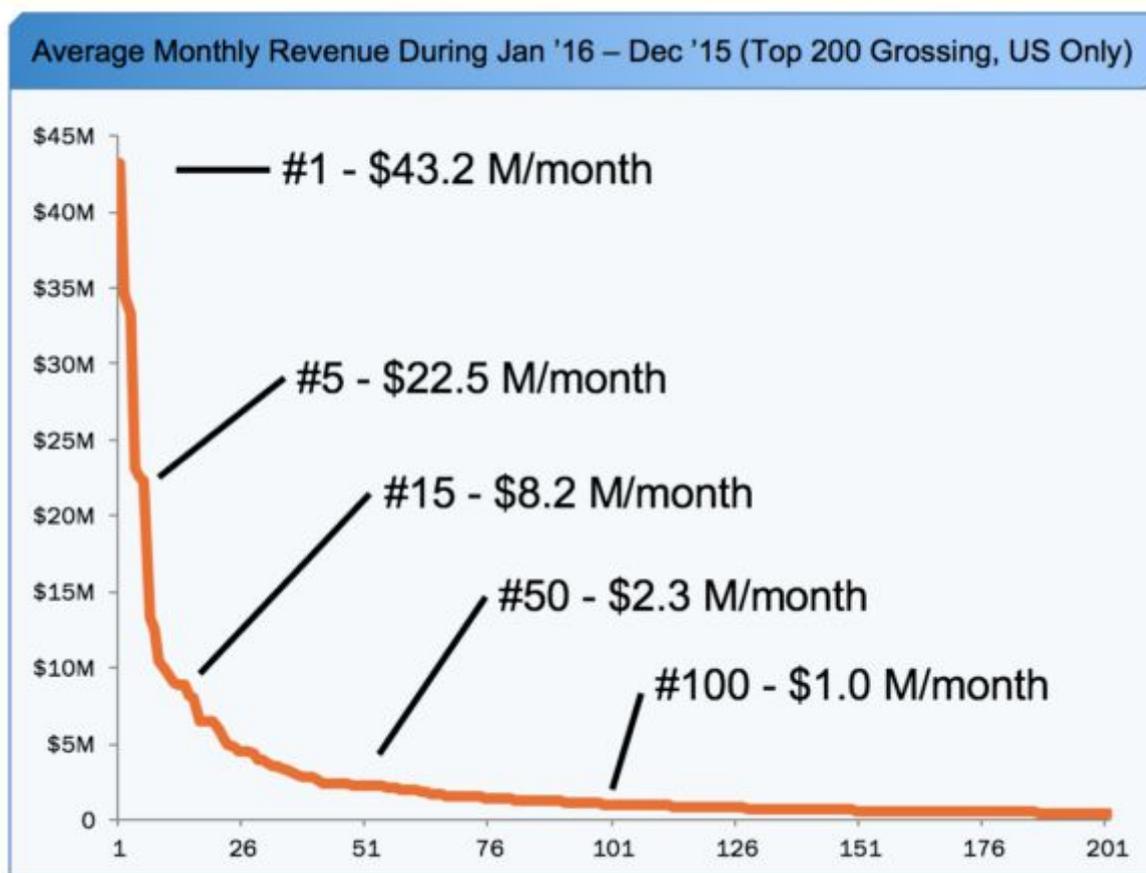


Figura 6: Jogos móveis por renda bruta total. O jogo número um nos Estados Unidos obteve uma média estimada de \$43.2M / Mês em renda bruta em 2016, comparado com apenas \$1.0M / Mês. para o jogo na posição 100.[3, 6]

Foi feita uma rápida entrevista com Túlio Caraciolo, da Manifesto Games, um estúdio fundado em 2005 especializado no desenvolvimento de jogos casuais e soluções educacionais lúdicas,[26] e algumas das dúvidas que são recorrentes neste tipo de mercado são, por exemplo, “qual o gênero de jogos que está caminhando para se tornar a próxima tendência?”, ou ‘como um gênero X está se saindo em relação a um gênero Y?’.

3.2. Acessibilidade a Business Intelligence

“Quanto dinheiro nosso jogo móvel irá faturar? Quando lançarmos este jogo, ficaremos ricos!” é um sentimento que é comum a grande maioria dos desenvolvedores de jogos móveis. Enquanto que milhões de dólares seria conveniente, na realidade, a maioria das pessoas ficaria satisfeita em conseguir desenvolver algo que entretesse o público e gerasse alguma renda mínima suficiente para que o processo se repetisse. Independentemente da intenção inicial, de querer ser gigante ou obter apenas o suficiente para viver, é sempre

prático se preparar previsões de receita. Criar um jogo custa muito dinheiro e tempo. Saber de antemão a renda potencial auxilia na avaliação de risco versus retorno do seu plano de negócio e permite que decisões otimizadas sejam feitas sobre o produto.[4]

Em geral, a maioria dos jogos móveis atualmente apostam na abordagem **Gratuito para Jogar (Free to Play, ou F2P em inglês)** com o modelo de **Software como um Serviço (software as a service ou SAAS em inglês)**. Enquanto títulos premium podem sem bem sucedidos, a grande maioria da renda do mercado de aplicativos móveis se dá a partir de microtransações devido à mentalidade dos jogadores, que tendem a gostar e a voltar para jogos em que as interações sejam frequentes e demandem pouco esforço. Enquanto que um título premium precisa que um jogador seja convencido previamente de que vale a pena que dinheiro seja gasto nele, títulos F2P prova para o consumidor de que aquele jogo é digno de se investir dinheiro e tempo e tenta mantê-lo engajado.[13]

4. Estado da arte

Dada a crescente necessidade de monitorar, entender e se adequar ao mercado de aplicações móveis, foi criada uma gama de ferramentas para este fim. Ferramentas com este propósito são comumente referenciadas como ferramentas de ASO.

Há muitos exemplos de ASOs no mercado, mas neste capítulo, apenas dois dos mais relevantes serão analisados, uma vez que, por serem os mais completos, já englobam todas as características que seus concorrentes menores possuem.

4.1. AppAnnie

App Annie[7] é uma companhia de dados e introspecção do mercado de apps móveis responsável por produzir informação sobre downloads, renda, avaliação, termos de busca e uso (MAU/DAU) para auxiliar em vários fatores relacionados a um app, como otimização de gastos de marketing, estratégias de investimento, parcerias e priorização do *roadmap* do app.[9]

O serviço online é composto por vários serviços, cada um focado em resolver uma determinada necessidade do mercado:

- Store Stats – Serviço online para análise de ranqueamento de apps. Provê, via gráficos históricos, dados sobre ranqueamento diário, notas e apps destacados, sendo possível fatiar os dados de várias formas, como país e categoria. Contem dados para apps de iOS, Mac, Google Play, Amazon Appstore, Windows Phone Store e Windows 8 Store.
- App Store Analytics – Um dashboard de vendas de apps para publishers que permite que se acompanhe vários dados sobre um determinado app, como quantidade de downloads, receita, posições no ranking e reviews dos usuários, tudo em múltiplas plataformas. É possível, além de acessá-los pela interface web, obter relatórios diários via e mail, exportá-los ou acessá-los a partir de uma API.

- Intelligence – Serviço que provê estimativas do mercado através de um algoritmo para auxiliar na estratégia de negócio dos apps.

Há muitos outros serviços, parte grátis, como analytics para a loja de eBooks e para Advertising, parte para usuários premium, como *Usage Intelligence*, mas que não serão abordados neste trabalho.[8]

App Annie provê, de forma gratuita, o serviço de Store Stats, Analytics e Advertising Analytics. O restante, no entanto, que envolve Business Intelligence, é adquirido através de um plano mensal.

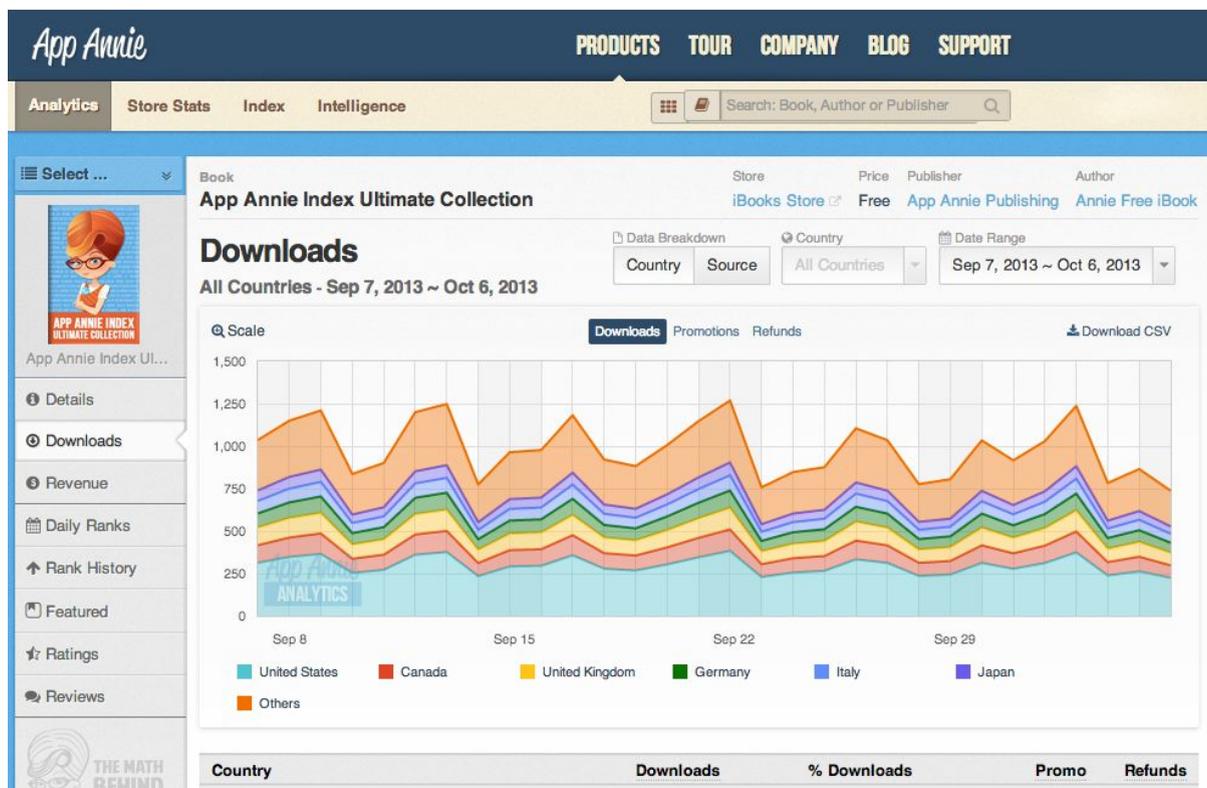


Figura 7: Dashboard de analytics de um dado aplicativo no App Annie.

4.2. SensorTower

Sensor Tower[10], assim como App Annie, é uma ferramenta que permite que desenvolvedores rastreiem e gerenciem diferentes palavras-chave na AppStore, focado em ASO e targeting geral de apps. A plataforma é dividida em duas partes: a primeira é uma seção grátis que provê ferramentas de palavras-chave e diferentes tipos de visualização de rankings; a segunda é uma seção paga com ferramentas de BI.

Assim como seu concorrente, Sensor Tower possui uma gama de funcionalidades, como histórico de ranqueamento de apps, histórico de ranqueamento de palavras-chave, quadros de liderança e top ranqueamentos. Além disso, ele também fornece uma estimativa de valor

de um dado app, utilizando uma série de diferentes fatores e métricas que incluem atividade dos usuários, alcance internacional, estimativas de receita, dentre outros. Como não há um padrão na indústria para este tipo de cálculo, é uma diferencial da plataforma. Com esta estimativa, a plataforma também provê uma métrica de estimativa de valor do publisher, agrupando o valor estimado dos apps para isto.

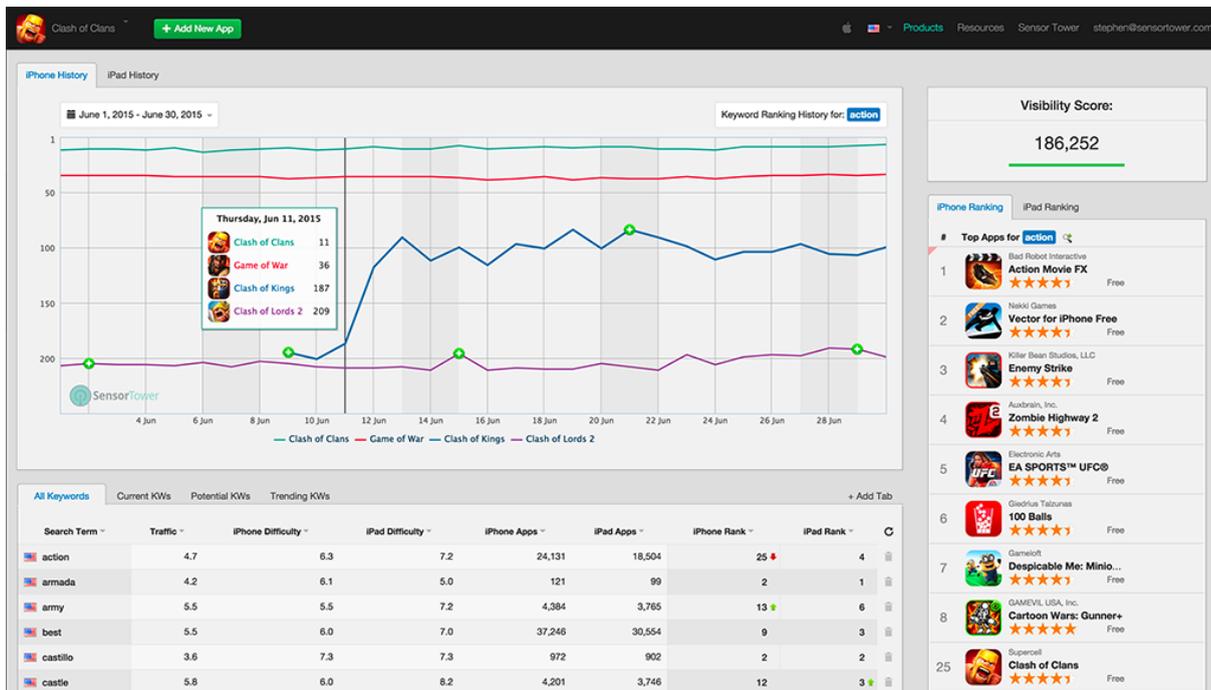


Figura 8: Dashboard de ASO do SensorTower. É possível rastrear categorias e palavras-chave de um dado app.

4.3. Comparativo

Abaixo, segue uma tabela com um comparativo das características principais dos ASOs descritos acima[11].

Característica/ASO	Sensor Tower	Appannie
Grátis/Teste grátis	Sim – limitado em 1 app (requer cartão de crédito para teste)	Sim (grátis com recursos limitados)
Suporte a iOS	Sim	Sim
Suporte a Android	Sim	Sim
Ranking de palavras-chave	Sim	Sim
Pesquisa por palavra-chave	Sim	Sim
Sugestões de palavra-chave	Sim	Sim
Espião de palavras-chave	Sim	Sim
Ranking por categoria	Sim	Sim
Rastreamento de competidores	Sim	Sim
Relatórios de instalação	Sim	Sim

Análise de notas/reviews	Sim	Sim
Relatórios via email	Sim	Sim
Internalização	Sim	Sim
Acesso a API	Sim	Sim
Visibilidade de pontuação	Sim	Sim
Exportação de dados	Sim	Sim
Integração com terceiros	Não	Não
Análise pré-lançamento	Não	Não
Sonar de aplicativos	Não	Não
Observações	Plano básico muito limitado	Ferramenta muito completa, mas pouco é acessível no plano básico
Keyword Accuracy	Alta	Alta

Tabela 1: Comparativo entre dois dos mais influentes ASOs do mercado atualmente: App Annie e Sensor Tower. A tabela leva em consideração os tipos de ferramentas que é normalmente fornecida por um ASO e analisa se o serviço em questão a possui.[11]

Enquanto muito completas, estas duas ferramentas demonstram uma característica muito comum entre os ASOs disponíveis no mercado: nenhum disponibiliza análise inteligente sobre os dados de forma gratuita, apenas através de planos pouco acessíveis para pequenos desenvolvedores. Além disso, ainda há pouco ou nenhum suporte para análise de pré-lançamento.

5. Proposta

Este capítulo detalha a proposta para a resolução dos problemas descritos no capítulo anterior.

5.1 Curvas de tendência sobre renda bruta

Este trabalho propõe a criação de uma ferramenta simples para a criação e visualização de curvas de tendência de jogos e de categorias de jogos em um determinado período de tempo a partir do seu posicionamento na PlayStore no decorrer do tempo.

Outro requisito da proposta é que o projeto deve ser de código aberto. Visto que sempre há barreiras monetárias para a obtenção do histórico dos aplicativos das lojas, um dos objetivos é tornar possível que qualquer um consiga, ao longo do tempo, construir sua própria base de dados.

Vale salientar que o objetivo não é competir nem substituir com nenhuma das soluções existentes, mas sim prover uma alternativa na obtenção dos dados, prover uma base para

que qualquer um possa trabalhar sobre os mesmos como desejar e explorar um novo tipo de visualização (pelas curvas de tendência).

Uma vez que o objetivo é extrair informações a partir das posições de um ou mais jogos em um ranking, foi preciso decidir alguma forma de visualização que condensa-se tal informação em um cenário onde dois ou mais jogos fossem ser visualizados simultaneamente. Em um gráfico com os dados de um jogo ou gênero, o olho humano consegue com certa facilidade encontrar um padrão nos dados, mas isto se torna mais difícil quando os dados de diferentes entidades começam a se misturar. Para tentar facilitar a separação dos dados durante a visualização, foi decidido criar curvas a partir dos dados disponibilizados, uma vez que elas, apesar de esconder parte da informação, a condensa de forma com que a tendência dos dados se mantém.

5.2. Método e fonte de obtenção dos dados

Os dados são obtidos da Google PlayStore. A Google não disponibiliza abertamente dados históricos detalhados sobre os aplicativos disponíveis na sua loja, portanto, para se obter esse histórico é preciso:

1. Obtê-lo manualmente, coletando-os diariamente;
2. Obtê-lo de terceiros que oferecem serviços de obtenção de dados antigos da PlayStore.

A maioria das ferramentas de ASO que existem atualmente já fazem este trabalho de manter sua base de dados atualizada, mas normalmente não disponibilizam estes dados históricos de maneira totalmente gratuita. O que é normalmente feito, como é observado nas plataformas do AppAnnie e do 42Matters[12], é disponibilizar estes dados brutos em conjunto com estatísticas e insights apenas para usuários pagantes, com opções de um período gratuito de testes (porém com permissões limitadas).

Dito isto, para este trabalho, com o objetivo de testar a ferramenta, parte dos dados foram obtidos a partir de uma das plataformas citadas anteriormente (42Matters). O restante fora obtido manualmente pelo script desenvolvido para ser executado diariamente pelo servidor para consultar os dados de determinados jogos da PlayStore e armazená-los no banco de dados da aplicação.

Para simplificar as consultas para a construção das curvas, os dados armazenados no banco consistem basicamente apenas da data de obtenção da métrica, do jogo em questão (assim como gênero) e sua colocação no ranking diário. Estes dados foram normalizados no banco de dados, ou seja, há duas tabelas: uma para os jogos e suas meta informações, outra para o ranking diário de cada jogo, com cada registro desta última apontando para um registro de um jogo da primeira tabela. No script de obtenção de dados diários, primeiro é feita uma chamada à api da playstore para se obter os cem primeiros colocados do dia da categoria GAMES. Em seguida, são feitas consultas para cada uma dessas posições no banco, checando se o jogo em questão já está registrado no banco. Caso não esteja, cria-se um registro novo na tabela de jogos. Em seguida, utiliza-se a chave do registro na tabela de jogos como chave estrangeira na criação do novo registro do ranking diário.

```
[ { url: 'https://play.google.com/store/apps/details?id=com.playappking.busrush',
  appId: 'com.playappking.busrush',
  summary: 'Bus Rush is an amazing running game for Android! Start running now!',
  developer: 'Play App King',
  developerId: '6375024885749937863',
  title: 'Bus Rush',
  icon: 'https://lh3.googleusercontent.com/R6hmyJ61s6wskk5hHFoW02yEyJpSG36i14JBkVf-Aojb1q4ZJ9nrGsx61wsRtn',
  score: 3.9,
  price: '0',
  free: false },
  { url: 'https://play.google.com/store/apps/details?id=com.yodo1.crossyroad',
  appId: 'com.yodo1.crossyroad',
  title: 'Crossy Road',
  summary: 'Embark on an action arcade, endless runner journey!',
  developer: 'Yodo1 Games',
  developerId: 'Yodo1+Games',
  icon: 'https://lh3.googleusercontent.com/doHqbSPNekdR694M-4rAu9P2B3V6ivff76fqItheZGJiN4NBw6TrxhIxCEpqq0',
  score: 4.5,
  price: '0',
  free: false } ]
```

Figura 9: Exemplo de json obtido através API da PlayStore contendo os dois aplicativos (em ordem decrescente, ou seja, Bus Rush em primeiro lugar, seguido de Crossy Road) com maior renda bruta total na data da consulta.

6. Implementação

Para a implementação do projeto, foi escolhida uma pilha de tecnologias web para facilitar e agilizar o desenvolvimento. O código fonte está disponível em um repositório aberto no Github[20].

6.1 Arquitetura

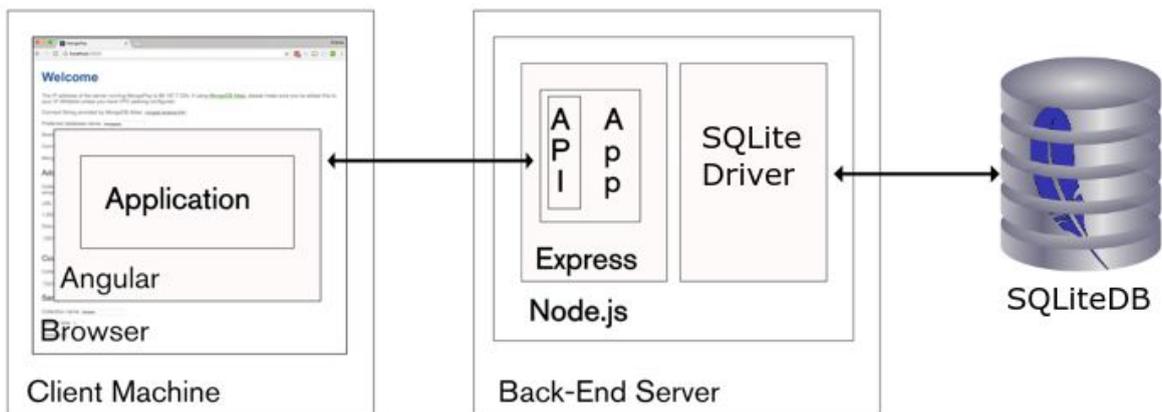


Figura 10: Arquitetura completa do sistema, demonstrado os componentes do cliente (Web) e do servidor.

O sistema foi dividido entre um cliente e um servidor (Figura 1). O servidor é responsável tanto pela coleta e pelo armazenamento dos dados de jogos da playstore. Já o cliente web é responsável por gerar as curvas de tendência para cara jogo/categoria, assim como providenciar uma interface de fácil uso para a visualização dessas curvas.

A arquitetura do sistema consiste de uma variação da pilha MEAN, porém utilizando SQLite via Sequelize em vez de MongoDB. Esta escolha foi feita visando o objetivo do sistema. Bancos NoSQL são amplamente utilizados em aplicações web, já que boa parte delas não é tão voltada para consultas. O sistema proposto, no entanto, beneficia-se mais com um banco relacional, visto que, até em sua forma inicial mais básica, é necessário o cruzamento frequente de dados. Caso fosse um sistema para mostrar apenas dados de um certo jogo, que não se relacionasse com o restante, um banco NoSQL seria uma alternativa viável, mas como são necessárias consultas temporais e agrupamentos diversos, tabelas normalizadas são mais adequadas e menos redundantes.

O restante da pilha mantém-se. Node.js é utilizado tanto no cliente como no servidor. Express é utilizado no servidor, blabla. Angular é utilizado no cliente, blabla.

6.1.1. Servidor

O servidor tem como função principal gerenciar os dados brutos obtidos dos jogos da Playstore. Isto inclui atualizá-lo a diariamente, armazená-los e disponibilizá-los via uma API, que será consumida principalmente pelo cliente web.

Os endpoints disponíveis na API do sistema estão listados na tabela abaixo:

Descrição	URL	URLParams	QueryParams
Obter lista de jogos	/api/games/	-	-
Obter dados de um jogo específico	/api/games/:id	:id - ID do jogo desejado	-
Obter posições do ranking de um determinado jogo	/api/games/:id/placements/	:id - ID do jogo desejado	start_date: data de início end_date: data de fim
Obter lista de gêneros	/api/genres/	-	-
Obter posições do ranking de um determinado gênero	/api/genres/:id/placements/	:id - ID do gênero desejado	start_date: data de início end_date: data de fim

Tabela 2: Lista dos endpoints do sistema. OS endpoints se resumem a **/games** e **/genres**, tendo cada um acesso a seu próprio **/placements**.

6.1.2. Cliente web

O cliente web é responsável tanto por permitir a visualização e customização das curvas a partir dos conjuntos de dados selecionados como por construir as curvas a partir dos dados recebidos pelo servidor.

Para a construção a curva de cada jogo ou gênero, é utilizado um algoritmo genético (como explicado no capítulo de referencial teórico) sobre a lista das posições diárias no ranking de maior renda bruta. Como o algoritmo espera um conjunto de coordenadas (X, Y) para construir as curvas sobre um plano cartesiano, é necessário sempre tratar os dados de entrada, mapeando as datas, em ordem crescente, no eixo Y.[17]

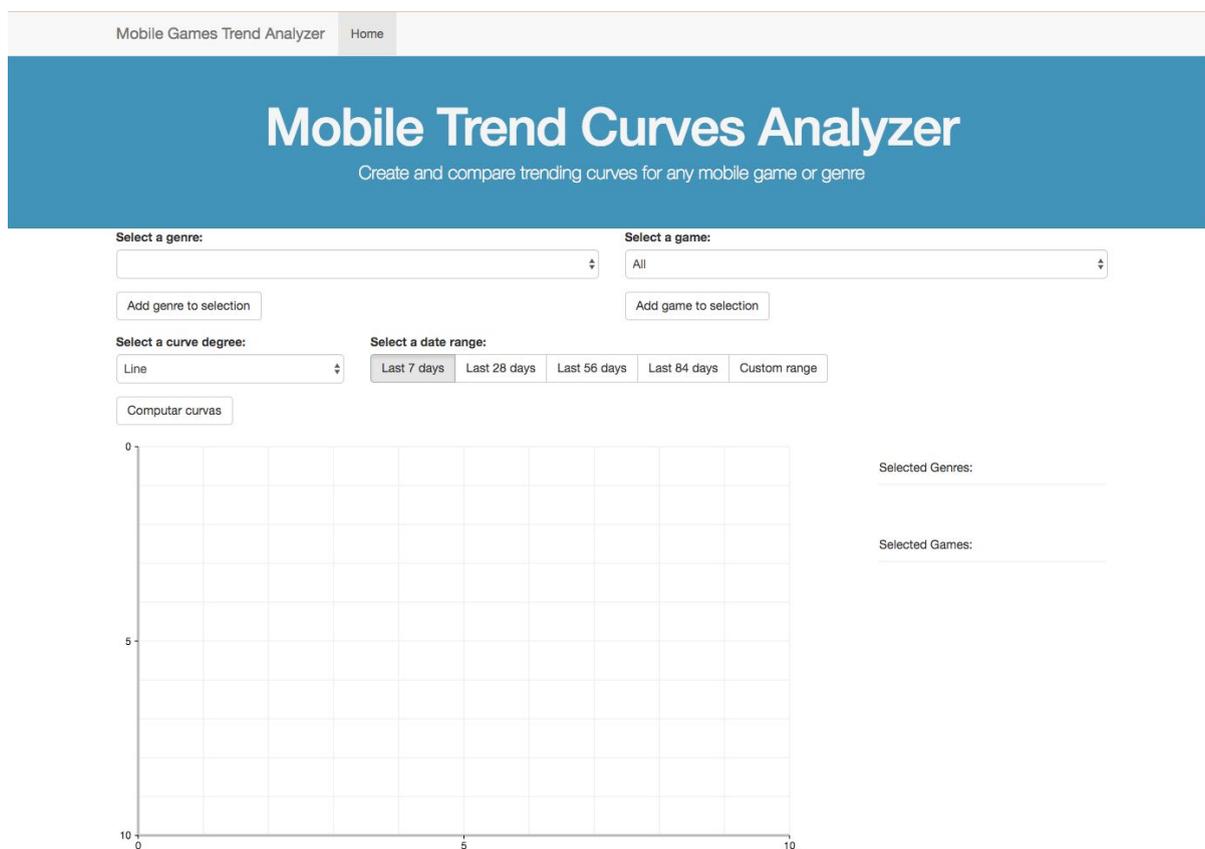


Figura 11: Página inicial do sistema web.

Por exemplo, caso a lista de posições recebida tenha entradas da data 01/01/2017 até 31/01/2017, cada data é mapeada de forma que 01/01/2017 torna-se $Y = 0$, 02/02/2017 torna-se $y = 1$ e assim sucessivamente, até que 31/01/2017 torna-se $Y = 31$. Desta forma, o gráfico não depende das datas em si, mas apenas da quantidade de dias entre as datas final e inicial.

Com os dados mapeados corretamente para o formato de coordenadas (X, Y), os mesmos são utilizados como entrada para o algoritmo genético responsável por criar as curvas. O

algoritmo visa otimizar os coeficientes de um polinômio para encaixá-lo num conjunto de dados (pontos no plano cartesiano). Dado um polinômio $ax^4 + bx^3 + cx^2 + dx + e$, os coeficientes a, b, c, d, e são gerados aleatoriamente a cada criação de curva e, em seguida, passam pelo ciclo de criação de novas gerações de soluções, ou seja, são selecionados iterativamente, depois passam por mutações e, em seguida, são cruzados por um dado número de gerações (500 nesta implementação). O cruzamento dos dados usa apenas uma interpolação entre dois valores dos pais selecionados. A quantidade de coeficientes depende da escolha do usuário quando a quantidade de curvas que ele quer gerar. Quanto maior o grau da curva, maior a quantidade de coeficientes necessários. Por exemplo, para a geração de uma linha, apenas, são necessários apenas os coeficientes d e e [Figura 12].

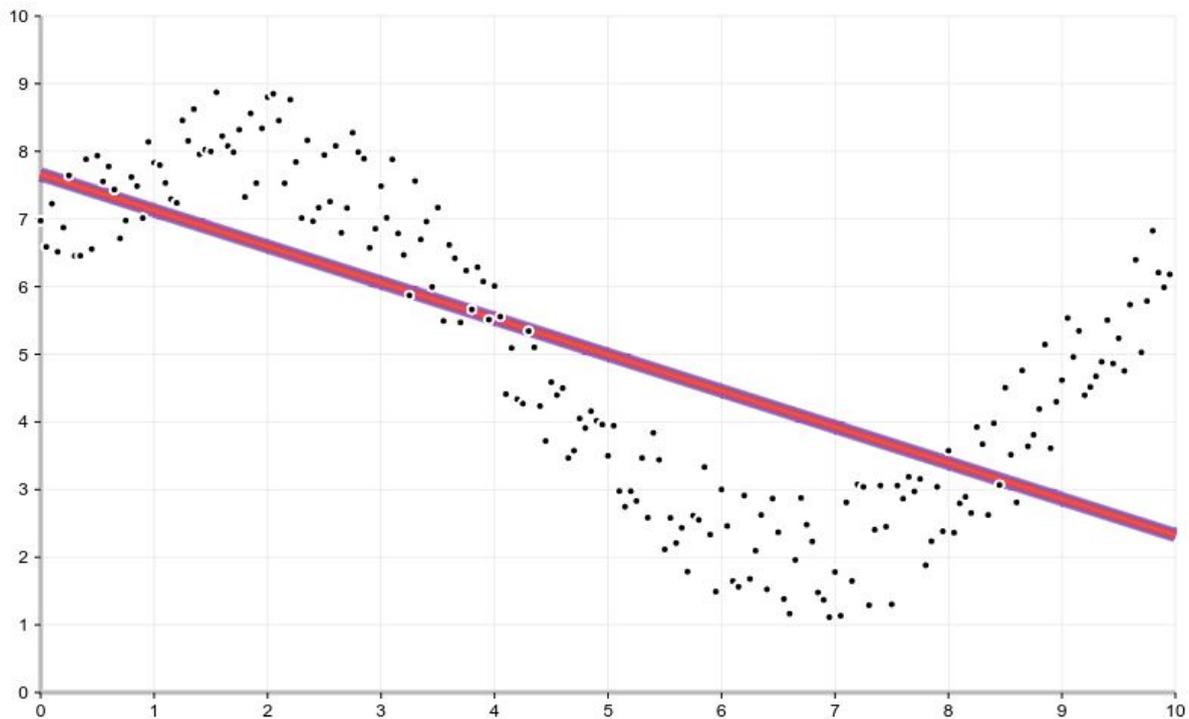


Figura 12: Exemplo de linha computada a partir do algoritmo genético utilizado para o ajuste de curva.

Para a criação de cada uma das curvas, é possível customizar os seguintes atributos:

- Tipo de curva: Linear, cúbica ou polinomial;
- Jogo ou categoria de jogo;
- Data de início e fim da análise.

Figura 13: Tela de customização da criação das curvas de tendência.

Após a seleção dos jogos e gêneros desejados, das datas inicial e final e do tipo de curvas, o cliente envia requisições para o servidor, afim de obter os dados relativos a cada jogo e categoria selecionados. Com isto, são executados N web workers, no qual N é a quantidade total de jogos e gêneros selecionados, calculando a curva para cada um deles.

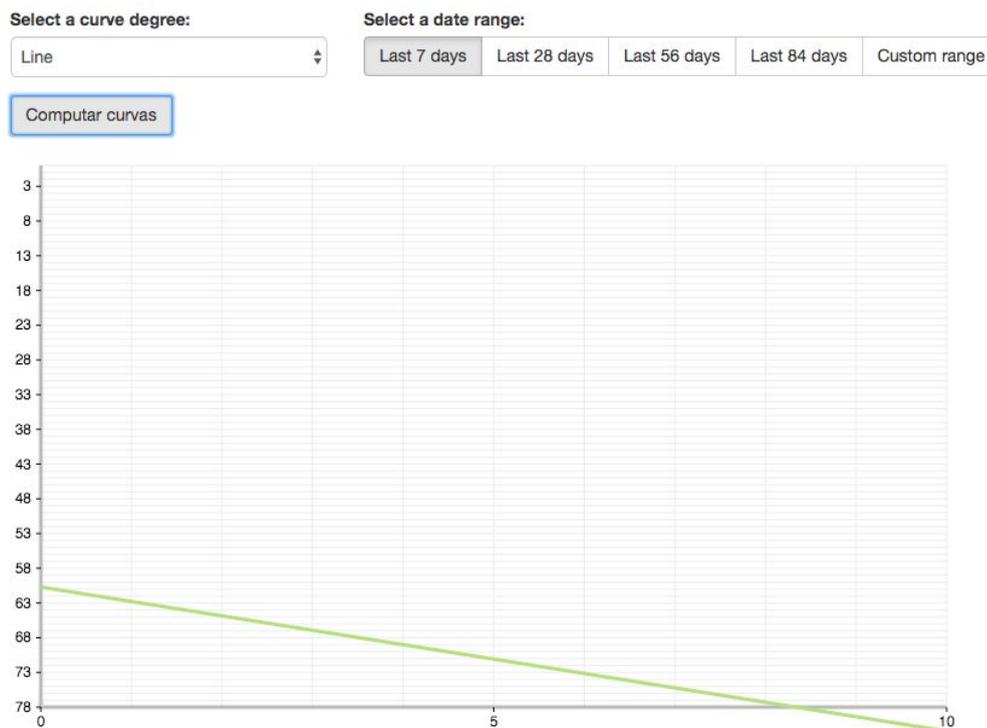


Figura 14: Linha gerada pelo sistema para um jogo selecionado

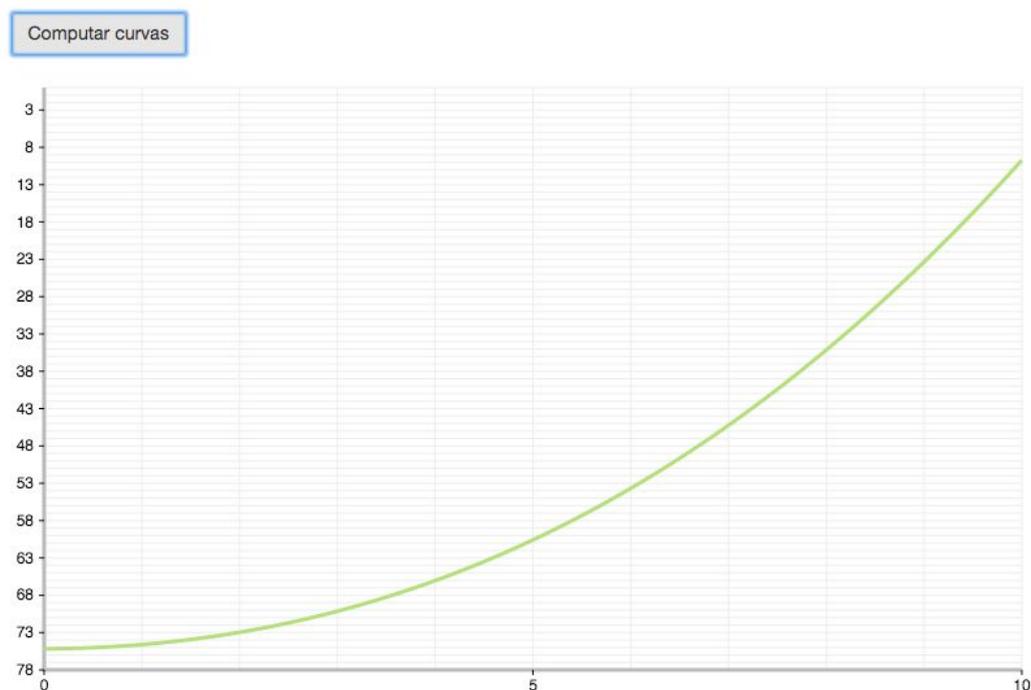


Figura 15: Curva gerada pelo sistema para um jogo selecionado

6.2. Resultados

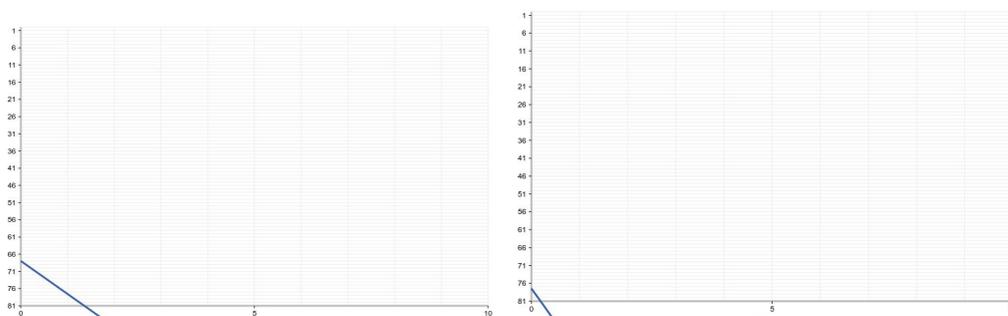
Com esta geração de curvas básica a partir de jogos ou gêneros, apesar de ainda precisando de vários refinamentos, principalmente na usabilidade, já torna possível a visualização do andamento de fatias do cenário de jogos móveis.

As imagens a seguir são de gráficos utilizando a base de dados inicial de teste (disponível na pasta “*seeds*” no projeto disponível online[24]). Vale salientar que os testes a seguir foram executados 20 vezes, com os resultados sendo praticamente iguais a cada execução.

Tomando como base o jogo **Bowmasters**, do gênero aventura, considerando os dados dos últimos 28 dias (de acordo com a data do teste), são gerados os gráficos da [Figura 16]. Neste período, as posições do jogo vão de 68 até 83 (sendo 1 a posição mais alta do ranking, e 100 a mais baixa), caindo uniformemente até o último dia. O primeiro gráfico (**Line**) e o segundo (**Curve**) demonstram bem isto: a curva mostra um decrescimento no ranking, passando até dos limites visuais do gráfico (que está limitado, atualmente, até a posição 100). É importante salientar que as curvas em si não indicam a posição final do jogo no ranking, mas apenas indicam, possivelmente, sua tendência com o passar do tempo, e a intensidade com que seu crescimento ou decaimento tende a mudar.

Os resultados mostram que, dos três tipos de curva (linear, parábola, polinômio), apenas o primeiro acaba se tornando relevante de alguma forma, já que ele mostra um estado imediato a partir das informações utilizadas como entrada. A curvas de parábola e polinomial sempre forçam um comportamento exponencial, então elas acabam sempre indicando que algo crescerá ou decairá muito.

Além disso, também é possível visualizar a fatia de dois ou mais jogos ou gêneros simultaneamente, com o intuito de facilitar comparações. Isto é demonstrado na [Figura 17], que mostra, em conjunto com o jogo utilizado no exemplo anterior, **Last Day on Earth: Survival**, outro jogo vindo da base de testes. Ele exhibe um comportamento aparentemente oposto ao do Bowmasters. Enquanto, no primeiro gráfico de linha, ambos demonstram um decrescimento, suas situações diferem nos gráficos seguintes. Considerando alguns fatores a mais, o gráfico de curva sugere que o segundo jogo tende a ganhar um pouco de ranking, mas o último gráfico sugere que talvez o ganho não seja tão provável e que sua posição tende a se manter, talvez até cair.



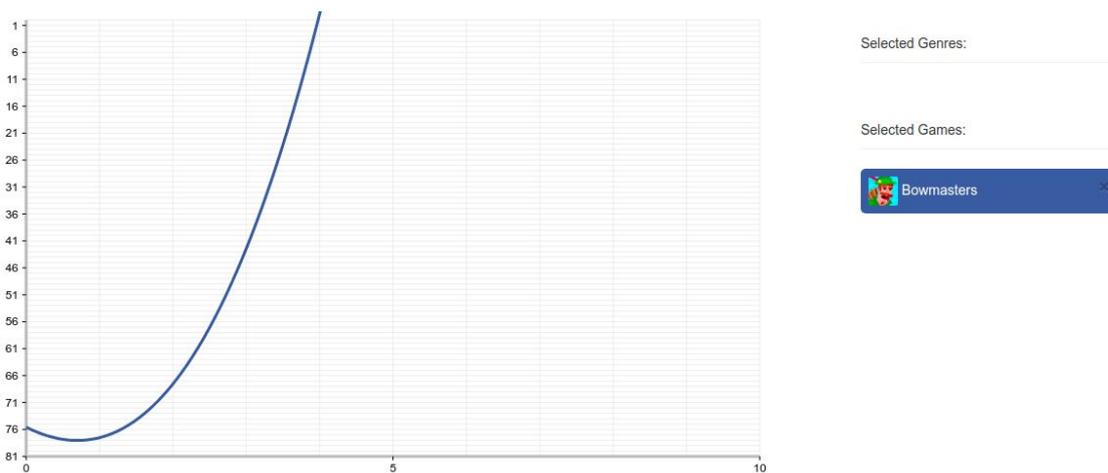


Figura 16: Gráficos de tendência gerados a partir do jogo **Bowmasters** a partir dos dados dos últimos **28 dias** (utilizando os dados de teste) utilizando os graus **Line** ou polinômio de primeiro grau (esquerda superior), **Curve** ou polinômio de segundo grau e **Polynomial**, utilizando polinômio de terceiro grau.

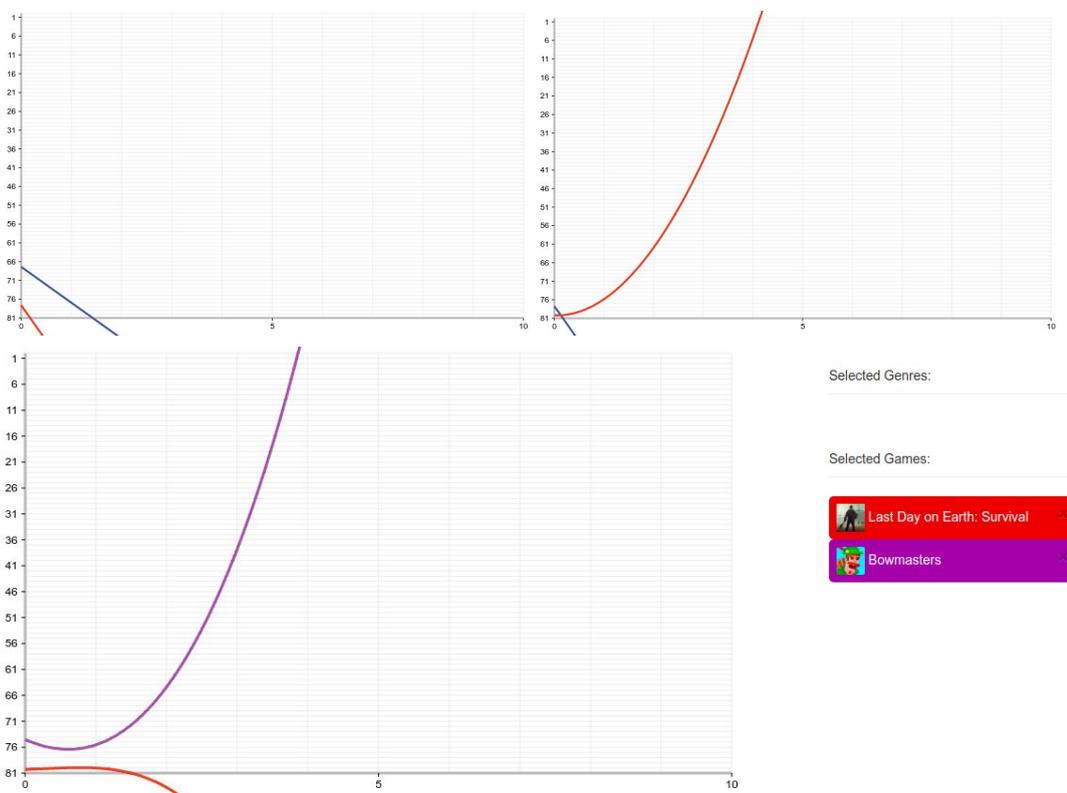


Figura 17: Gráficos de tendência gerados a partir do jogo **Bowmasters** em conjunto com **Last Day on Earth: Survival** a partir dos dados dos últimos **28 dias** (utilizando os dados de teste) utilizando os graus **Line** ou polinômio de primeiro grau (esquerda superior), **Curve** ou polinômio de segundo grau e **Polynomial**, utilizando polinômio de terceiro grau.

7. Conclusões

Este trabalho propôs uma ferramenta simples de código aberto, disponível no github, para a obtenção de dados de jogos da PlayStore e criação de curvas de tendência a partir dos dados de renda bruta de cada jogo ou categoria de jogo. É possível se notar que ainda há uma gama de tipos de análise inteligente que pode ser feita sobre esse tipo de dado, e esta ferramenta tem como propósito servir como base para tal.

A ferramenta implementada ainda é muito imatura, apesar dos resultados dos testes serem consistentes com as múltiplas execuções, ainda há certa incerteza sobre a qualidade das informações geradas pelas interpretações das curvas. É preciso, com o passar do tempo, validar melhor se as informações inferidas condizem com a realidade, trazem algum valor, e ir ajustando o algoritmo de acordo com isso.

Além disso, pode-se observar que, uma vez que todas as configurações de curvas, exceto a linear, forçam um comportamento exponencial, elas não são tão boas e efetivas como uma interpolação livre, sem seguir uma equação.

Com a maioria das ferramentas de BI fora do alcance da maior parte da comunidade não pagante, é importante a criação de pequenas iniciativas para que os dados e suas inferências sejam o mais abertos e disponíveis possíveis.

Para trabalhos futuros, pretende-se resolver algumas pendências de implementação da solução atual, assim como validar melhor e expandir as funcionalidades básicas da ferramenta. Isto inclui:

- Criar testes automatizados;
- Melhorar a documentação do código;
- Criar um sistema básico de autenticação;
- Validar com desenvolvedores a viabilidade e o interesse sobre as funcionalidades providas - atualmente, ainda há certa incerteza sobre o valor gerado para o usuário final;
- Flexibilizar o mecanismo de obtenção de jogos;
- Expandir para obter dados da AppStore;
- Testar outros tipos de análise sobre os dados de renda bruta ou outras métricas;
- Flexibilizar e criar mais customização quanto à construção de curvas de renda bruta (feedbacks indicam que seria bom, por exemplo, poder misturar os tipos de curvas).
- Melhorar os controles de customização de curva e expor mais informações sobre os dados selecionados;
- Utilizar uma ferramenta mais robusta para exibição e construção dos gráficos, permitir melhor visualização dos pontos e curvas geradas fora do alcance atual (100 posições), ajustar o eixo x para indicar fielmente a quantidade de dias, imprimir, além das curvas, as posições do jogo no gráfico, uma vez que, a curva por si só, condensa e esconde parte da informação.

8. Referências

[1] Kim, J. (2017). 2016 Mobile Gaming Trends Report: Top 8 Take-aways. [online] Gamasutra.com. Disponível em: http://www.gamasutra.com/blogs/JosephKim/20170427/296933/2016_Mobile_Gaming_Trends_Report_Top_8_Takeaways.php [Acessado em 1 Jul. 2017].

[2] Filho, Vicente Vieira, Atila Valgueiro Malta Moreira, and Geber Lisboa Ramalho. "Deepening the Understanding of Mobile Game." 2014 Brazilian Symposium on Computer Games and Digital Entertainment(2014)

[3] Brightman, J. (2017). Mobile games booming as global games market hits \$108.9B in 2017 - Newzoo. [online] GamesIndustry.biz. Disponível em: <http://www.gamesindustry.biz/articles/2017-04-20-mobile-games-booming-as-global-games-market-hits-usd108-9b-in-2017-newzoo> [Acessado em 30 Jun. 2017].

[4] https://www.gamasutra.com/blogs/RossPrzybylski/20170213/291280/How_Much_Money_Will_Your_Mobile_Game_Make.php

[5] Sonders, M. (2017). New mobile game statistics every game publisher should know in 2016. [online] Medium. Disponível em: https://medium.com/@sm_app_intel/new-mobile-game-statistics-every-game-publisher-should-know-in-2016-f1f8eef64f66 [Acessado em 3 Jul. 2017].

[6] Newzoo (2016). GLOBAL GAMES MARKET REPORT. [online]. Disponível em: https://cdn2.hubspot.net/hubfs/700740/Reports/Newzoo_Free_2016_Global_Games_Market_Report.pdf [Acessado em 4 Jul. 2017].

[7] App Annie. (2017). App Annie - The App Analytics and App Data Industry Standard. [online] Disponível em: <https://www.appannie.com/en/> [Acessado em 3 Jul. 2017].

[8] App Annie. (2015). Mobile Games: Now You Can Predict the Future. [online] Disponível em: <https://www.appannie.com/en/insights/app-adoption-cycle/> [Acessado em 4 Jul. 2017].

[9] App Annie. (2017). Mobile Games: Now You Can Predict the Future. [online] Disponível em: <https://www.appannie.com/en/insights/app-adoption-cycle/> [Acessado em 3 Jul. 2017].

[10] - Sensor Tower. (2017). Sensor Tower - Mobile App Store Marketing Intelligence. [online] disponível em: <https://sensortower.com/> [Acessado em 04 Jul. 2017].

[11] Theapprepublic.com. (2017). Sensor Tower Review | App Republic. [online] Disponível em: <http://theapprepublic.com/sensor-tower-review/> [Acessado em 04 Jul. 2017].

[12] AG, 4. (2017). 42matters: Mobile App Intelligence & Audience Data. [online] 42matters.com. Disponível em: <https://42matters.com/> [Acessado em 10 Dez. 2017].

[13] Klipfolio.com. (2017). What is Mobile BI?. [online] Disponível em: <https://www.klipfolio.com/resources/articles/what-is-mobile-business-intelligence> [Acessado em 04 Jul. 2017].

[14] SG, C. (2017). 10+ ASO tools to help with Google Play Store Optimization for your app. [online] Androiddb.com. Disponível em: <http://www.androiddb.com/2014/07/10-tools-to-help-with-google-play-store-optimization-for-your-app/> [Acessado em 04 Jul. 2017].

[15] Apptentive. (2017). 9 Top App Store Optimization Tools - Apptentive. [online] Disponível em: <https://www.apptentive.com/blog/2017/02/28/9-top-app-store-optimization-tools/> [Acessado em 04 Jul. 2017].

[16] En.wikipedia.org. (2017). Curve fitting. [online] Disponível em: https://en.wikipedia.org/wiki/Curve_fitting [Acessado em 04 Jul. 2017].

[17] GitHub. (2017). subprotocol/genetic-js. [online] Disponível em: <https://github.com/subprotocol/genetic-js> [Acessado em 04 Jul. 2017].

[18] Linden, R. (n.d.). Algoritmos Genéticos. 2nd ed. Brasport, p.40-43.

[19] Boomtrain. (2017). Measuring Retention Right: A Progression of Retention Methodologies. [online] Disponível em: https://boomtrain.com/measuring-retention-right-a-progression-of-retention-methodologies/?utm_source=quora [Acessado em 04 Jul. 2017].

[20] Business Model Generation, Alexander Osterwalder, Yves Pigneur, Alan Smith, and 470 practitioners from 45 countries, self-published, 2010

[21] International Business Times. (2017). Freemium Games Make Up 80% Of \$10B Mobile App Market In 2012: Flurry Report. [online] Disponível em: <http://www.ibtimes.com/freemium-games-make-80-10b-mobile-app-market-2012-flurry-report-936872> [Acessado em 04 Jul. 2017].

[22] Salesforce.com. (2017). What is Software as a Service (SaaS) - Salesforce.com. [online] Disponível em: <https://www.salesforce.com/saas/> [Acessado em 04 Jul. 2017].

[23] Distributed Application Architecture (PDF). Sun Microsystem. Acessado em 10 Dec. 2017.

[24] GitHub. (2017). tuliolages/mobile-games-trend-analyzer. [online] Disponível em: <https://github.com/tuliolages/mobile-games-trend-analyzer> [Acessado em 04 Jul. 2017].

[25] Sqlite.org. (2017). About SQLite. [online] Disponível em: <https://www.sqlite.org/about.html> [Acessado em 10 Dez. 2017].

[26] Manifesto Games. (2017). [online] Disponível em <http://www.manifestogames.com.br/site/home.php?lang=pt> [Acessado em 10 Dez. 2017].