



3 ステップで始める AI モデル開発！

始める前に知っておくべきこと！



Sheng Hsia Leng

アマゾン ウェブ サービス ジャパン合同会社
ソリューションアーキテクト

中村 一樹

アマゾン ウェブ サービス ジャパン合同会社
ソリューションアーキテクト

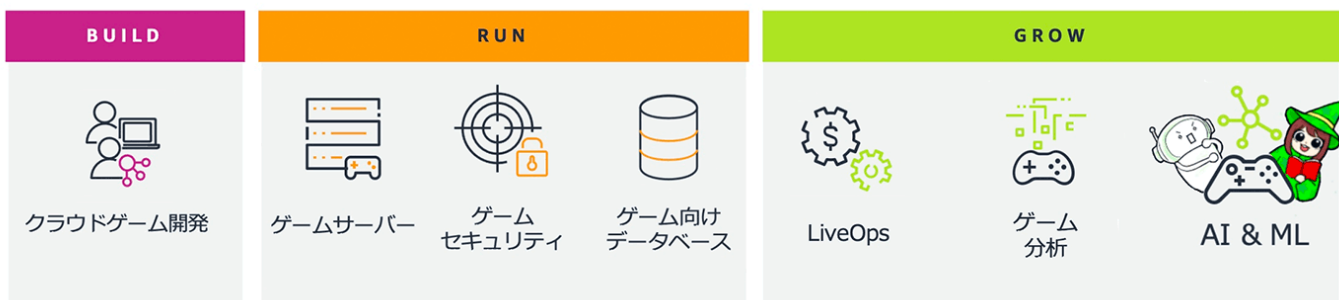


ゲームみなさんこんにちは、Game Solutions Architect の Leng と Game Solutions Architect の Kazuki です。

この投稿では [前回の記事](#) でゲームへの AI/ML の活用イメージはなんとなくついたが、結局どうやって始めれば良いかわからない、と言った方々の疑問や不安を少しでも取り払って AI/ML をゲーム開発や運営に役立てることができるようになることを目指した記事になります。よろしければぜひ [前回の記事](#) もご覧いただき、こちらの内容をご参考にしていただければ幸いです。

AWS FOR GAMES

[AWS for Games](#) ではより早い開発、よりスマートな運営、そしてより楽しいゲームへの成長という **Build**、**Run**、**Grow** の 3 つの柱に沿ってサポートします。本日は **Grow** の柱、AI&ML の入門の部分になります。



何から始めれば良いのか

前回の記事で紹介した「Amazon Kendra」や「Amazon Comprehend」などの AI サービスは既に学習済みですぐに使用できる AI モデルを使用しているため、比較的簡単に AI 活用を進めることができます。しかしサービスとして用意されていない分野の AI 活用を行いたい場合は自分たちで AI モデルを開発する必要があります。自分たちで AI モデルを開発するとなると、サービスを利用するよりは難易度が上がります。

また AI モデル開発を行ったこともないので、「何を準備して、どのように開発をしていけばよいかかわからない」ということもあると思います。AI モデル開発といっても比較的簡単にできるものから複雑で難解なものまで幅広いです。しかしほとんどの場合、基本的に決めなければならないことや工程に関してはそれほど大きく変わりません。せっかく AI モデルを開発するのであれば、開発成功率をあげ、AI 活用自体も成功させてゲーム開発や運用を

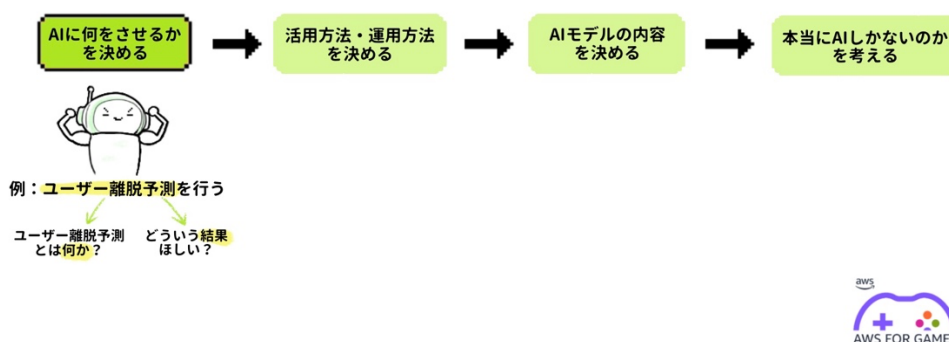
良くしていきたいでしょう。そこで今回は活用方法の一つ「ユーザー離脱予測」をベースに「何を準備して」「どのように作っていくのか」を3つのステップで解説していきます。

ステップ 1：作り始める前に様々な要件を決める

このステップ1では AI モデルを作る前に決めなければならない様々な要件を決めていきます。この工程は AI モデル作成において最重要といっても過言ではありません。どれだけ詳細に一つ一つの要件を決めるかによってこの後の工程や AI モデル開発、活用が成功するかどうかを左右します。

ステップ 1-1：AI に何をさせるかを決める

まずは「AI に何をしてもらい、どうなればよいのか」ということを決めます。自動プレイをさせる、ゲームをやめてしまいそうなユーザーを予測する、などの要件です。当たり前のようなことですが、この内容をしっかり決めておかなければ、作るものがはっきりせず開発期間が伸びてしまったり、目指していたモデルから少しずれたモデルが完成してしまったりします。その結果「この AI モデルは使えない」「AI の活用はうまくいかない」「ただの技術検証になってしまった」といった評価になってしまいます。ステップ 1-1 をしっかり定義していないことで、本来は AI 活用が有効なケースにも関わらず「AI 活用自体が悪い」という結論になりかねません。このようなことを起こさないためには「AI に何をさせたいか決める」ということはとても重要です。



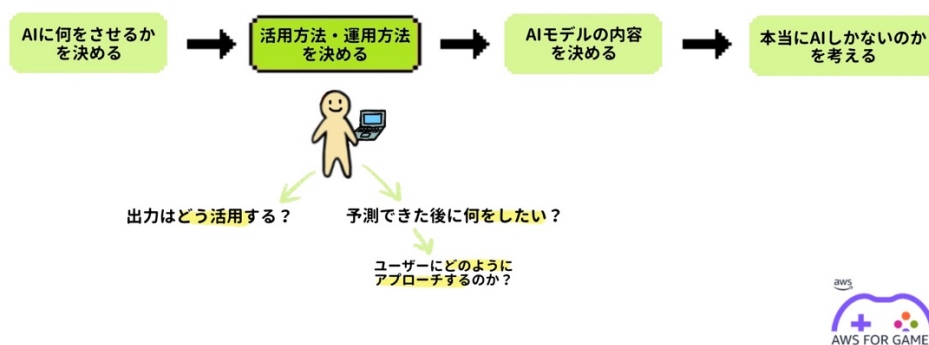
今回の場合ユーザー離脱予測という案件なので「ユーザー離脱予測を行う」ということを決めます。その後「ユーザー離脱予測とは何か」を決めていきます。「ユーザー離脱予測」とはゲームをやめてしまいそうなユーザーを、ユーザーのプレイログなどのユーザーデータからどのユーザーがやめてしまうのかを予測してピックアップする、といったものです。この定義は最低限ではありますが、AI に何をさせてどういう結果が欲しいか、ということがわかるレベルにはなっています。

ステップ 1-2：活用方法、運用方法を決める

次に活用方法を決めていきます。これが決まっていなければどれだけ優れた AI モデルが完成しても宝の持ち腐れになってしまいます。AI モデル開発がただの技術検証に終わってしまうケースの多くはこの活用方法を明確にしないまま AI モデル開発を始めてしまったことが原因です。

AI モデルはなんらかの入力に対して、なんらかの出力を返します。その出力がどのようなもので、何に活用するか、といった AI モデル自体の活用方法をまずは決めます。

出力自体の活用方法が決まったら次は「どのように運用していくか」を決めていきます。AI モデルは更新するのか、AI モデルはどれくらいの頻度で実行するのか、手動でやるのか、自動化するのか、といった運用方法も決める必要があります。この運用方法まで決めておかなければ AI モデルが完成してもゲーム運用に利用することができず、1度も実践投入されずに終わってしまう、といった結果になってしまいかねません。そのため活用方法と運用方法を決めることもとても重要です。



このことを今回のテーマの「ユーザー離脱予測」で考えてみます。ユーザー離脱とは運営中のゲームでプレイ中のユーザーがなんらかの原因でやめてしまうことです。このユーザーの離脱の予測ができたら運営側は何ができて、何をしたいのかを決めていきます。予測なのでまだそのユーザーはやめていない状態なのでそのユーザーに対してなんらかのアプローチを行うことで離脱を阻止することができるかもしれません。しかしこのアプローチについては慎重にならなければなりません。安直に「離脱しそうなユーザーに対してなんらかの優遇処置をとる」といったことを行った場合、このシステムを悪用される、あるいは他のユーザーとの間に不公平が生まれてしまうことにより、本来は継続してくれていたユーザーのモチベーションを下げ離脱させてしまうといったことを起こしてしまう可能性があります。そのため予測できた後にそのユーザーにどのようにアプローチをするか、といったことはこの段階である程度明確にしておく必要があります。

活用方法には実際に AI モデルを使用する以外にもう一つあります。それは AI モデル分析です。AI モデル自体がどのような特徴を重視して判断をしているか、といったことが特徴量重要度などからわかります。この特徴量重要度を分析することにより「どのようなことがユーザーの離脱につながっているか」といったことがわかるようにな

ります。これによりゲーム内容を改善することで離脱ユーザーを減らすこともできるかもしれません。このように AI モデル自体の活用とその AI モデルを分析することで得られる情報の活用、この二つを決めておきます。

次に運用方法についても決めていきます。バージョンアップごとにモデルを更新する、1 日 1 回特定の時間にユーザーデータを AI モデルに渡し、離脱ユーザーを予測させ、ピックアップリストを出力する。そのピックアップリストをもとにユーザーにアプローチを行う、といったようにどのようにゲーム運営に AI 活用を組み込んでいくか、をあらかじめ決めておきます。このようにあらかじめ活用方法と運用方法を決めておくことで AI モデル完成後にすぐに実践投入することが可能になります。

ステップ 1-3 : AI モデルの内容を決める

ステップ 1-1 とステップ 1-2 が決まると次は AI モデル自体の内容を細かく決める必要があります。内容とはどれくらい賢いか、どういうことをさせるのかの詳細などです。例えば自動プレイの場合は「どのくらいのレベルのユーザーと同じような自動プレイをさせるのか」といったことであり、何かを分類するものであれば「何と何を分類させるのか」といったものです。ここで決めるものは AI モデル開発のゴールを決めるものになります。ここを明確に決めておかないの AI モデル開発を終わらせることができなったり、よくない AI モデルが出来上がったりしてしまいます。

今回のテーマは「ユーザー離脱予測」です。予測しなければならないものは「離脱ユーザー」つまり「ゲームをやめてしまったユーザー」です。この「やめてしまったユーザー」とはどのようなユーザーなのでしょう。毎日プレイしていたユーザーがなんらかの理由で 1 日だけゲームをしなかったらそれは離脱ユーザーになるのでしょうか。何日連続してゲームをプレイしない日が続いたら離脱ユーザーになるのでしょうか。「やめてしまったユーザーの定義」人それぞれだと思います。離脱してしまったとする日数には答えはないので AI モデルを開発する側が定義する必要があります。例えば、1 週間 (7 日間) のうちにアクセスが 1 日以下のユーザーを「離脱ユーザー」として定義します。

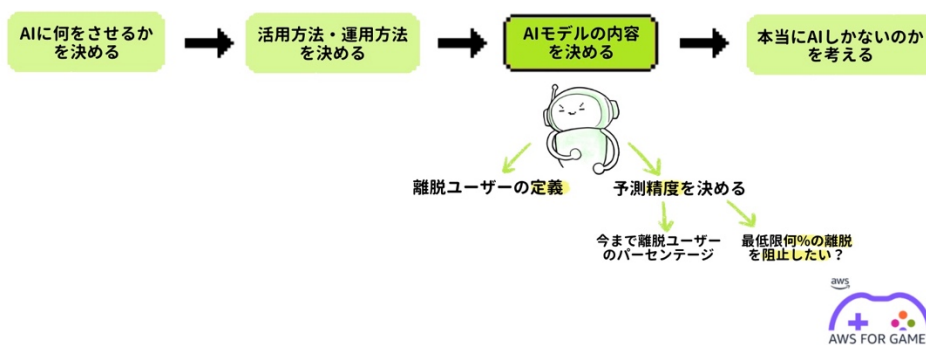
離脱ユーザーの定義ができた次は予測精度を決めます。この予測精度というのは AI の賢さにあたるものになります。作るからには 100% の精度が望ましいですがそれは不可能です。

そのため「やりたいことをするためにはどれくらいの精度が必要なのか」を決める必要があります。もちろん 100% に近づくことはとても良いことではありますが、精度を上げることばかりを行っているといつまでも AI モデルを完成させることができません。しかし完成させるためにあらかじめ精度を低くしていてもそれは使えない AI モデルを開発することになるので意味がありません。やりたいことに応じて適切な精度を設定する必要があります。

医療など人の命などに関わるの分野に関しては限りなく 100% に近づけることに価値もあり、その必要性もあります。精度 80% や 90% といったものでは使い物にならないかもしれません。その一方で中には 60% 程度の精度で良いとされているものもあると思います。この 60% でよい案件に関しては 60 以上の精度が出ていれば 70 や 80 にしていく工程はそれほど重要でなく、無駄なコストになるかもしれません。このようにやりたいことを実現するために必要最低限の精度もとても重要なものになっています。

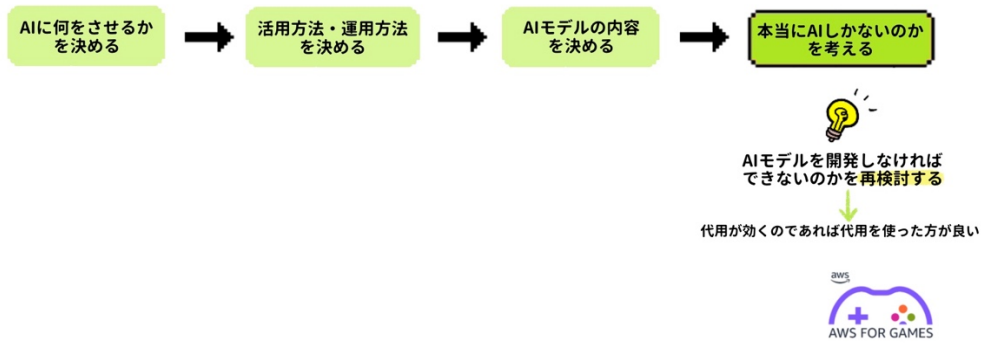
ユーザー離脱予測でこの精度を考える場合は「どの程度の精度でユーザー離脱を検知できれば目的を達成できるのか」という観点で考えます。今までの離脱ユーザーのパーセンテージやその中の最低限何%の離脱を阻止したいか、というものなどから実際の精度を決めていきます。「現状は新規ユーザーの 80% がすぐに離脱しているので離脱率を 70%以下にしたい」と言う場合、今より 10% 離脱率を下げる必要があります。次に何らかのアプローチを行うことによって予測したユーザーの 20% が残ると仮定します。そうすると「80 (現在の離脱率) x (必要な精度) x 0.2 (アプローチをして残る数 20%) = 10 (下げたい離脱率)」という計算式ができ、これを計算すると「0.625」となるので最低限 62.5% 以上の精度が必要となります。

ユーザー離脱予測に関しては「離脱ユーザー」と「精度」この 2 つが定義できれば準備は完了です。



ステップ 1-4：本当に AI 活用しか方法がないのかを考える

今までのステップで AI モデル開発に必要な定義の準備が整いました。そこで最後に「本当にやりたいことは AI モデルを開発しなければならないのか」を再検討します。やりたいことなどが今までのステップで定義されているので十分に検討できる材料は揃っています。中には統計的な処理や分岐文で済むものもあるかもしれません。多くの場合、AI モデルを実用レベルにまで持っていくには数ヶ月かかってしまいます。決して簡単に試して成果が出るというものではありません。代用が効くのであれば代用を使った方が良いでしょう。



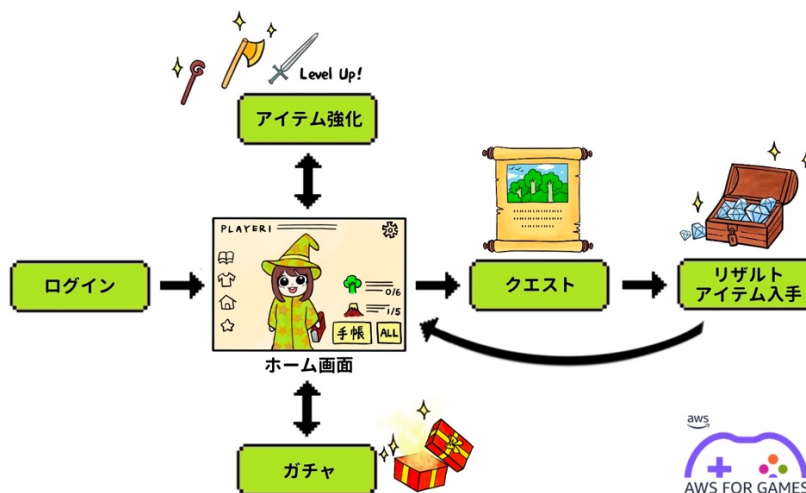
AI モデル開発はやりたいことをやるための手段でしかありません。やりたいことがより簡単に行える方法があるのであればそちらを使うべきです。そのため最終的にエンジニアに「AI でなければやることができないか」といったような相談を行うことをオススメします。

ステップ 1 のまとめ

ステップ 1-1~1-3 は AI モデルを開発する以上に重要なものです。このパートの内容が明確にすることができなければ AI モデル開発をすることはオススメできません。決めないまま開発を始めると、ほとんどの場合失敗に終わってしまいます。AI モデルの開発は決して低いコスト、短時間で開発できるものではありません。また開発しても目指した精度に届く保証もない不確実もないものです。しかしそれらのリスクを背負っても試す価値が AI にはあると私は思います。このステップ 1 で作り始める前に決めなければならないものは決まりました。次のステップでは作るために必要な材料、「データ」の準備を行なっていきます。

ステップ 2 に入る前に

ステップ 2 以降は簡単なゲームのイメージがあった方が話を進めやすいので以下のようなゲームをイメージで話を進めていきます。




ステップ 2：前処理、データ準備

ステップ 2 では AI モデル開発に必要なデータを準備していきます。AI モデルを作成するためには AI に学習させるための教材であるデータが必要になります。どれだけ良いデータを集められるか、加工できるかが AI モデルの精度を左右するものになるので、AI モデル開発工程内ではとても重要な工程になります。

ステップ 2-1：必要なデータを集める

AI モデルの学習に必要なデータを集めていきます。今回はユーザー離脱予測なので、ユーザー離脱に関係がありそうなデータを集めていきます。必要なデータは主にユーザーの行動ログになります。行動ログの中からユーザー離脱に影響を与えていそうなデータを集めていきます。



ログサンプル (アクセスログ)


userid	created_at	api	type	userlv
magician1	2023-06-01	login	pc	27
demon777	2023-06-01	additem	mobile	10
tensaiplayer	2023-06-01	queststart	mobile	150
awesome123	2023-06-01	login	pc	3
elasticplayer	2023-06-01	gacha	pc	54
secondplay10	2023-06-01	gacha	mobile	35
blacksword	2023-06-01	questend	pc	128
⋮				

しかし行動ログは膨大でその中から何を集めれば良いかわからなくなることもあると思います。データ収集方法は人それぞれで多々ありますが今回はその中の一つの「仮説」に基づいてデータを集めていくことにします。

仮説とは文字通り「このようなことが起こった場合ユーザーは離脱する」といった仮説を立ててその仮説を立証するために必要なログを集めていきます。例えば「難しいクエストに何度も挑戦して失敗している人は離脱している」といった仮説の場合、クエストのクリア率やクエスト失敗数などクエストに関わるデータが必要になります。この他にも「ガチャを引くためのアイテムがなくなってしまった場合」「クエストを全てクリアしてしまいやることなくなくなってしまった場合」といったように様々な仮説を立てて、その仮説に基づく「アイテム数」や「クエストクリア数」などのデータを取得していきます。

この仮説を立てるためには対象のゲームに関して詳しくなければなりません。そのため詳しい開発者を集めて多くの仮説を出して、信憑性の高いものから順にデータを集めていくことが良いと思います。

次に集めたデータだけではすぐに使えないものもあります。例えばクエストクリア数などはクリアしているクエスト ID のユニーク数をカウントする必要があります（こちらの図の "q_start_counts" など）。このように集めたデータを加工して新しいカラムとして追加して使用します。これらの集めたデータと加工したデータを 1 行にまとめ学習用データを作っていきます。




加工済みデータサンプル

userid	continuous_days	q_start_counts	sum_item_num
magician1	7	48	127
demon777	1	0	0
tensaiplayer	21	284	540

ステップ 2-2：正解ラベルをつける

各データを集め、加工し 1 行にまとめ終わると次に正解ラベルをつけていきます。正解ラベルとはこのユーザーが離脱したかどうか、を 0 と 1 で学習用データにつけていく作業になります（こちらの図の "churn_flag"）。



加工済みデータサンプル

userid	continuous_days	q_start_counts	sum_item_num	churn_flag
magician1	7	48	127	0
demon777	1	0	0	1
tensaiplayer	21	284	540	0

この数値を予測する

ここでステップ 1-3 で定義した「離脱ユーザーの定義」を使用します。例えば、1 週間のうち 1 日以下のログインユーザーを離脱ユーザーとする、という定義を立てた場合、その条件に当てはまるユーザーの正解ラベルに 1 をつけ、当てはまらないユーザーに 0 をつけていきます。今回のユーザー離脱予測 AI モデルは学習後、この正解ラベルについて予測していくことになります。

ステップ 2-3：学習用データと確認用データに分ける

ステップ 2-2 までで学習用データの準備は完了しました。最後に出来上がったデータを実際に学習に使うためのデータと精度確認の答え合わせを行う用のデータに分類します。

全体のデータ量次第にはなりますが、AWS のサンプルの場合は、学習用データ：検証用データを 7：3 の割合で分割していることが多いです。分け方に関してはランダムで分けても問題はありませんが、どちらのデータにも全種類のデータが偏りなく入っている必要があります。今回の場合は離脱ユーザーかそうでないユーザーかを予測するので正解ラベルの 0 と 1 の両方が含まれているデータを準備する必要があります。(自動で学習用データと精度確認用データを分けてくれるライブラリなどもあります)

学習用データと精度確認用のデータに分けることができたならデータの準備は完了です。

ステップ 2 のまとめ

AI モデルは学習用データに適応 (Fit) するように学習していきます。そのため AI モデル開発においてデータ準備がほとんどの作業を担っているといっても過言ではありません。優れた AI モデルは良いデータから学習され、悪いモデルはデータ数が足りない、データの種類の少ない、作成したデータが悪いといったことがほとんどです。

今回は「ユーザー離脱予測」をテーマにしているので予測のデータ準備について紹介してきましたが、その他の分野の AI モデル開発においても学習用データの重要性は何ら変わりありません。このステップ 2 が完了するといよいよ最後の工程の AI モデル自体の学習を行なっていきます。

ステップ 3：学習させてモデルを作成する

ステップ 3 では AI モデル開発の最終工程の学習させるパートに入ります。ステップ 2 で用意したデータを学習させて実際に活用できる AI モデルを作成していきます。

ステップ 3-1：学習方法を決める

学習用のデータ準備が完了すると、そのデータを利用して AI に学習させていきます。学習方法は様々な方法があり、自分で自作する方法もあれば、AutoML ライブラリを利用する方法もあります。特に AutoML ライブラリを使用する場合は、学習用データと精度確認用データを指定し、簡単な設定を行うだけで様々な学習方法やデータの

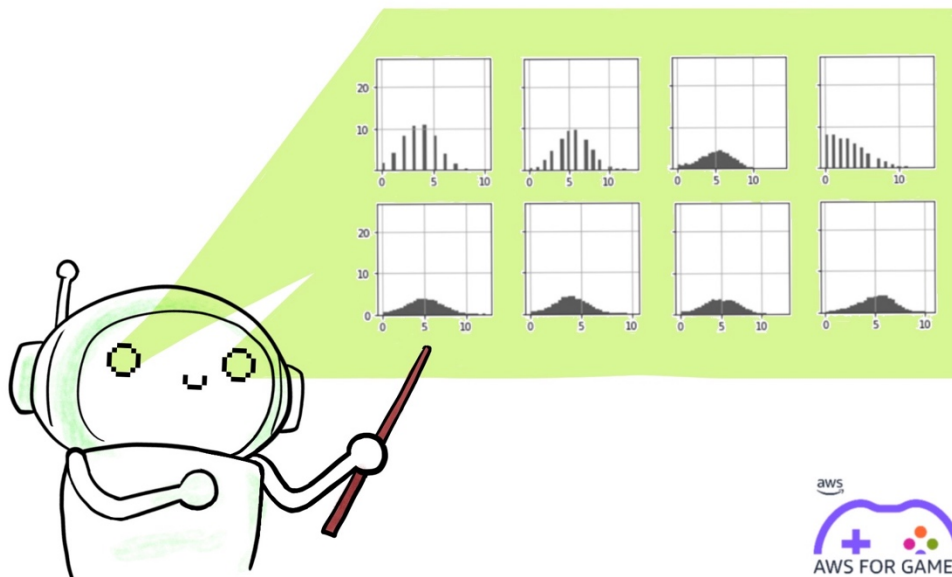
選択を自動で試し、一番良い結果の AI モデルを出力してくれます。AutoML ライブラリで実現できる活用方法であれば積極的に使用していく方がより早く活用、運用のフェーズに移行することができます。

Amazon SageMaker には最初から多くのビルトインアルゴリズムを提供しています。その中には上記の AutoML ライブラリも含まれています。どの方法で学習させ AI モデルを出力させるかを決め AI モデルを出力させます。

ステップ 3-2：精度を確認する

出来上がったモデルの精度を確認していきます。ステップ 2-3 で分けた精度確認用のデータを出来上がったモデルに与えて予測させ、その予測結果と正解ラベルを比較して精度を確認していきます。(AutoML ライブラリを使用している場合は学習段階で精度確認用データも渡すことで併せて精度確認してくれるものもあります) この精度がステップ 1-3 で決めた精度にとどいていればモデルの完成になります。

まだ求める精度に届いていない場合は予測するために必要なデータが足りない、仮説が間違っている、ということになるのでステップ 2 に戻り、仮説の検討やデータ集めからやり直しをします。1 からやり直すことは大変なので多くの場合、特徴量重要度などの AI モデルがどの数値に重きを置いて予測をしているかなど様々な数値を確認できます。この数値を参考にし、残すデータ、変更するデータ、追加するデータを検討していきトライアンドエラーを繰り返して精度を高めていきます。



ステップ 3 まとめ

AI モデル開発においてどのように学習させるか、といった方法は AutoML ライブラリの登場により自動で選択させることが主流となってきています。そのため AI モデルは開発においても初学者が参入しやすくなってきています。AI モデルが完成すればあとはどのように活用するか、運用させるか、になります。作っただけでは宝の持ち腐れなので [ステップ 1-2](#) で決めた内容をもとにまずは運用を開始していきましょう。

まとめ

簡単にですが AI モデルを自分で作っていく場合にはどのような検討や準備が必要でどのように開発していくかを解説していきました。今回はユーザー離脱をベースにお話しましたが各ステップでの必要な内容はどの AI モデルを作る場合にも参考になると思います。

今回紹介した内容が皆さんの今後の AI/ML 開発や活用のサポートになれば幸いです。