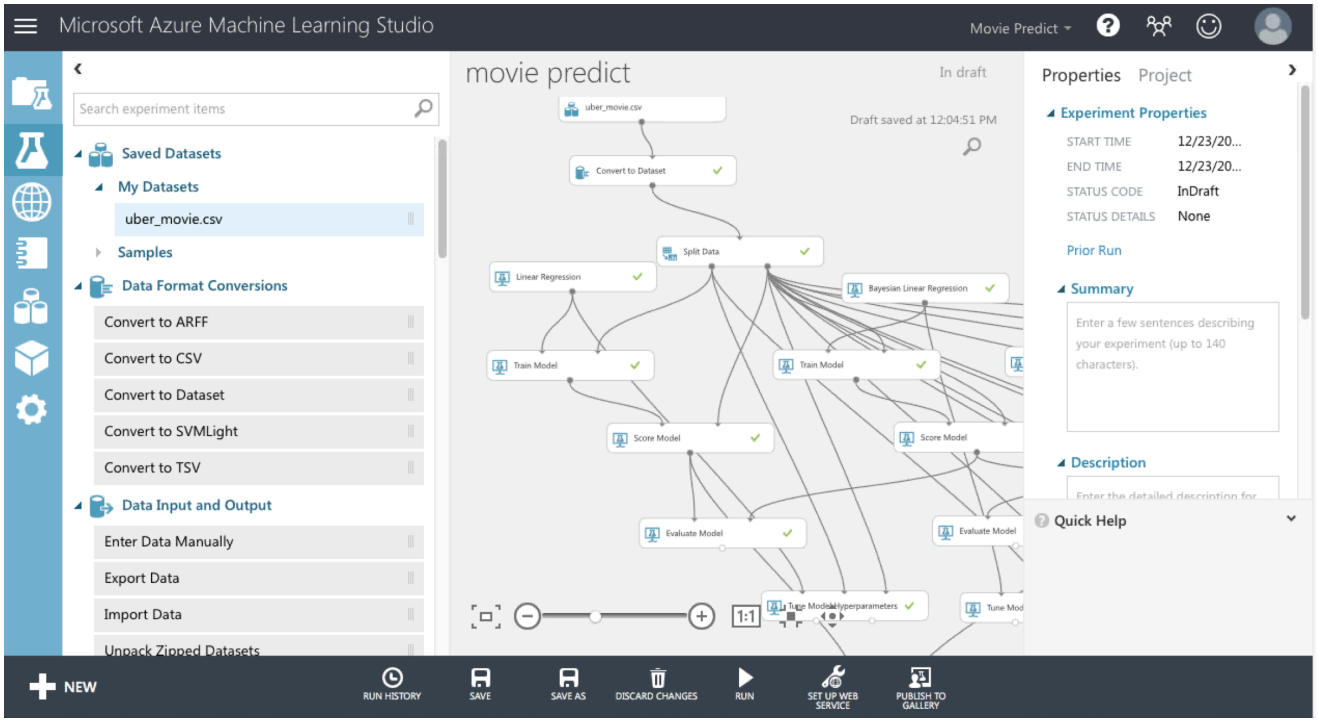
MACHINE LEARNING IN THE CLOUD

A Case Study for Predictive Analytics

February 2016



Move Fast

This paper provides tips and tools for machine learning in the cloud. It is intended for startups and fast moving small teams working under significant time and resource constraints. It reviews the process and tools used to evaluate the feasibility of developing a predictive analytics product for the entertainment industry that predicts box office revenues and the optimal date to release a movie.

Define the MVP

We defined the scope with some high level User Stories, many whiteboard diagrams and a narrative description. We called our MVP Movie Lineup.

Define the Benefits

Use Movie Lineup to select the release date with the most revenue. Manage your schedule, keep track of the competition and make the best choice when you need to change a release date. Movie Lineup allows you to set triggers to track when a competitor movie changes the date or when you reach the timeline threshold for one of your own properties.

Use the Movie LineUp dating release platform from The Big Data Company to select the release date with the most revenue. Manage your schedule, keep track of the competition and make the best possible choice when you need to change a release date. Movie LineUp allows you to set triggers to track when a competitor changes their release date or when you reach the timeline threshold for one of your own properties.

Why Data Science?

Data science can be used to track various inputs and dimensions, both historical and real-time, to provide on-demand predictions for the best release date. Input variables include global events calendar, studio schedules, competitive research, industry knowledge, market intelligence, social listening, probability speculation and undisclosed titles. These variables have different impact depending on a film's genre, cast, rating and target demographics. Each input is weighted and prioritized for each film and then data analysis is performed to determine the optimum date.

Define the Dataset

The dataset includes industry data (competition, financials, cast, contractual constraints, theater count, international releases), crowdsourced data (Twitter, YouTube), time series data (shifting production schedules, release dates) and international revenues and production budgets. There is a notable lack of production budget data, which is one of the primary features used for analysis and prediction. This was the primary challenge in developing effective algorithms.

Given this extensive, constantly updated dataset with significant gaps, we decided to approach the problem by time boxing the scope and quickly prototyping a solution based on out-of-the-box, easily accessible data science tools. Once the initial prototype is developed, the project will be re-evaluated for feasibility and market interest.

Identify the Team

Our team included a Product Manager, Data Scientist, UX Designer and Executive Stakeholder. This bare bones approach meant that each team member fulfilled multiple roles:

- The Product Manager developed specifications and requirements and performed project management.
- The Data Scientist wrangled the data, deciphered the data dictionary, imputed missing values, experimented with algorithms and developed the models.
- The UX Designer was responsible for front-end development, graphic design and user experience design for an interface with a series of input forms, sliders, dropdowns, state dependent messaging and sophisticated visualization reports based on the user's inputs/selections.

Twelve Week Sprint

We gave ourselves a timeline of three months to develop a Minimum Viable Product.

Business Problem (2 Weeks)

First we defined the business problem with executive stakeholders from the industry's top studios. We conducted question and answer sessions and stakeholder interviews. We developed user personas and developed rough mockups developed to present the product concept.

Data Import and Prep (2 Weeks)

The dataset was identified and prioritized. Partnerships were developed to obtain access to structured industry data via APIs. Crowdsourced data was collected from various publicly available platforms (IMDB, Rotten Tomatoes, YouTube, Twitter).

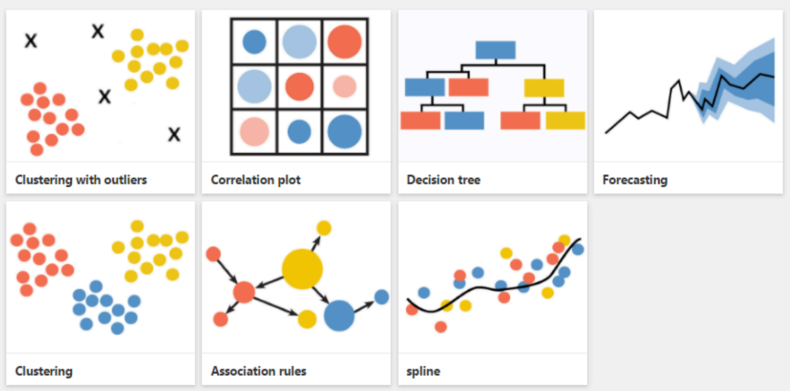
Cloud resources were configured in both AWS and Azure. We used AWS for compute, storage and database resources. We used Azure for out-of-the-box machine learning algorithms.

Data Engineering (1 Week)

Data was scrubbed, wrangled, joined and imputed to prepare a dataset suitable for training the models. This included a combination of data wrangling and feature selection from 103 variables.

Data Science (4 Weeks)

Training data was fed into various models. Various approaches and algorithms were employed to determine the accuracy of the results and therefore the feasibility of the product. The models were tested and scored.



Data Visualization (1 Week)

Data was visualized in interactive charts and graphs. Results were validated against the training dataset and selected real-world examples, including anomalies like Deadpool.

Interactive Prototype (1 Week)

An interactive prototype was developed in parallel with data visualization. The prototype allowed users to input movie parameters and return a movie lineup that displays competitor movies and box office revenues.

Go / No-Go (1 Week)

Results were presented to stakeholders, industry experts and potential clients to determine whether additional resources would be allocated to develop the product.

Our Biggest Challenge

We were missing data for the one feature with the most influence on our success: production budgets. We dev

eloped a model to estimate a film's production budget based on cast (directors, producers, actors, casting) and technical credits (special effects). This allowed us to estimate the production budget and radically improved predictions.

Tools and Services We Used

We are cloud agnostic, however, most of our projects and clients use AWS. For this project we used both AWS and Azure. Each service had pros and cons for this specific project.

AWS was used only for the EC2 instance for cloud computing power via RStudio, an Interactive Development Environment (IDE) for the programming language R. The EC2 instance delivered significant improvements in RAM and computing power not available on a desktop or laptop computer. This compute power was used to quickly make progress in examining datasets, wrangling data, building models, visualizing relationships between data variables, feature selection and feature engineering.

Azure ML is a Graphical User Interface Studio that allowed our data scientist to "point & click" his way to multiple models in minutes. It handled data import, data, cleaning and wrangling, as well as feature selection, training datasets, modeling selection, imputation, scoring the model and tuning parameters. Azure ML made quick work of deploying machine learning models on multiple datasets with ease and delivered metrics and results faster than coding and tuning the parameters manually. The downside was the metrics had to exported, and the visualization opportunities, while robust, were limited to what was available in the studio.

Azure

- Great for machine learning models, algorithms
- Great for data wrangling and imputing data
- Azure ML has a workflow and a visual editor that beginners can easily follow and build their first ML project with Azure ML
- Azure ML has common data cleaning and transformation tasks that you can use or you can also build the data pipeline using R code with Azure ML
- Azure ML supports more ML models than AWS:
    - Binary & Multiclass classification
    - Regression
    - Clustering
    - Recommendations
    - Anomaly detection

- For each problem, Azure ML gives you the option to try multiple algorithms — you can also bring other algorithms supported on R or IPyhton (or build your own!)
- Azure ML also helps you tune the parameters for each algorithm — in fact they have a “sweep parameter” task that iterates you multiple input options for each algorithm parameter and identifies the optimal parameter setting for your problem
- Azure ML also makes it easy to compare the performance of different algorithms and help you select the best one for the problem at hand
- It also supports R and Jupyter notebooks so you can port your existing R/Python code as well and use Azure Platform to operationalize your Machine learning project

AWS

- Used for EC2 compute power
- Amazon ML has a wizard that walks you through each step and so it enables developers to quickly get started
- Amazon ML gives you common performance metrics to evaluate your model’s performance — for example, if you are building a binary classification model then it gives you Binary AUC.

- Used for visualization using specific packages in RStudio such as GGPlot
- Used for plot colinearity experimentation
- QuickSight in-memory for data visualization
- Lacks out-of-the-box ML models (only two: regression and binary/multi-classification)

- No customization to tune ML models, no ability to select features or perform imputation, dataset must be clean, no feature selection
- Amazon ML supports basic data cleaning and transformation tasks — but you will have to do the heavy lifting of cleaning/transforming data somewhere else
- Amazon ML currently supports limited ML models:
    - Regression
    - Binary and Multi-class classification
- Amazon ML does not let the developer select the algorithm for the problem at hand — for instance, if you have a binary classification problem then it automatically uses Logistic Regression algorithm for you. It doesn't let you change the algorithm to something like Two-class SVM or Two-class decision forest

Other Tools

We used a variety of tools for this experimental project.

- Slack – instant communication, Q&A, problem solving
- Asana – task management
- Moqups – visual mockups
- QuickSight – data visualization
- AWS – compute
- Azure – machine learning
- Excel – share data with non-developers
- React and Redux – interactive prototype

- R - statistical programming language, selected over Python because it provides better data visualization capabilities
- Trifacta - agile tool for data manipulation (SCREENSHOTS), feature selection and feature engineering, “recipes”
- Knime / Azure - open source Azure, fast alternative to R to impute data SCREENSHOTS, select a module, tune the module, faster response/compute time, highly flexible
- Mice / Azure - superior to PCA (cluster analysis, but not a fit for the regression models)

Data Science is a Must Have

Data science is an emerging and rapidly developing field with applications that can be extended across the business. Use some of these tips to allow your data science team to quickly experiment and develop models that can be operationalized for your business.

THE BIG DATA COMPANY

The Big Data Company is part of the VentureSoft Global Group www.venturesoftglobal.com. We work with you to discover your most profitable products, customers and regions so you can determine where and when to cut costs, where and when to increase spend. You will have visibility into every aspect of your business so you can identify emerging markets, target spending and quickly respond to disruptors.

Authors

Alexander Raboin, Data Scientist

Christina Cheney, Data Analytics Strategy

www.TheBigDataCompany.us

@bigdata_club

info@thebigdatacompany.us

650-530-3282