During my current work experience at Deloitte Consulting, I was working on a project aiming to provide the client with an algorithm solution to improve the business performance. The client is in life insurance industry and they applied rate increase on their policyholders every year and they wanted to have an accurate prediction on the response from the policyholders once they informed the rate increase. I was one of the two folks in the modeling team and we started from understanding the business and brainstorming the data sources and target variables that could be leveraged. We identified few internal data sources such as the previous response information, policyholder demographic data and the claim data. Based on our experience, we proposed them with additional external data source from third party that could match the internal data on person level which could potential add value to the algorithm. The client sent the internal data to us and we matched and merged the external marketing data that eventually got our final modeling data with over 300,000 rows and 2400 columns (2GB). During the data wrangling process, I used SAS and SQL to manipulate different tables. One of the biggest issue we were facing was the data quality. Since the data was quite big, some observations were not able to prove what the client discussed based on business sense. So we spent a lot of time checking the data quality, going back and forth with the client asking for clarification. They ended up with some correction of the data and sent us the updated tables. The next thing we did was creating various synthetic variables based on the internal and external datasets that could potentially add lift to the model. For example, we created one variable that proved to be powerful in the following modeling phase, the affordability of the rate increase. We used rate increase from the internal data and household income from external data to create this specific variable.

Before the modeling phase was the “Exploratory Data Analysis” that we identified the correlation between each predictors against the target variable (the response type from the policyholder). We leveraged data visualization tool such as Tableau to show the segment chart between different groups and ended up identifying 500 highly correlated variables from 2400 variables that were considered to be predictors in the model.

During the modeling phase, we proposed the client with two options. One was the logistic regression which was easy to explain but hard to perform well compared to advanced techniques. The other was random forest model which has better predictions but less explainable. We ended up providing them both and each of the algorithm will be used in different business cases. For financial impact analysis which does not need to know the reason behind but requiring higher accuracy, random forest would be the direction to go. For figuring out the relationship between each predictors with the response and follow up with tailoring strategies, logistic regression would be the direction since it is much easier to quantify the relationship.

Given the tight deadline of the project, something that might have done better during the process. The biggest one is the feature engineering that currently we were using eye balling on the correlation as well as the segment chart to figure out the strong predictors and then used stepwise regression to filter out more variables. If there is enough time, we could try different advanced techniques such as Random Forest and GBM to see the variable importance output from each algorithms and compared different techniques to lock down the powerful predictors. Also there are more advanced feature engineering techniques that could be leveraged in this process.

1. One challenge was correctly leveraging the model stacking methodology and still keeping the model explainable. I was working on a life insurance project helping the client to predict the risk class for the life insurance applicants. To make the model explainable we were going with logistic regression model to identify applicants who have higher health risks and should go through medical tests before making a decision whether or not to give this person an offer. The problem with GLM model was the performance. I know the model stacking techniques could leverage the performance but we did not want to make the model hard to explain because the client was really interested to know the reason behind. I found a way to combine both by combining the result from decision tree to the existing GLM model. We built few decision trees and identified few buckets in the tree which were interested (had a higher risk prediction). Then we used these buckets as individual indicators in the GLM model. To avoid the overfitting problem, we did not select the final nodes from the decision trees but instead choosing some intermediate nodes. This technique proved to be very helpful during our modeling phases and these indicators were still explainable.
2. The other challenge was to identify the right level of the modeling dataset for analysis. I led an effort on predicting NCAA tournament result at Deloitte Consulting that we were predicting the probabilities of winning a game during the March Madness. We had datasets with information on historical regular season game results, historical tournament game results and historical seed information. The challenge was at what level we need to make this prediction and how. Since the game is about two teams’ performance so it did not make sense to predict team A’s winning probabilities solely. Also, in each year every team was performing differently so it made more sense to use “year+team” as an unique team rather than team solely. So I ended up rolling and merging all the datasets to the game level with all the Team 1 information along with Team 2 information. The next challenge was using which part of the data as the modeling dataset. Using regular season game result may not help a lot so I ended up using historical tournament data as the training set but included regular season information for each team as predictors. The last challenge was how to find the strong predictors. What we had was the team level information but the game level prediction really depended on the comparable information from the two teams. So I created various synthetic variables showing the ratio and differences between the two teams and in the final model result, most of the strong predictors were from this creation such as the seed difference between the two teams and the average scoring difference between the two teams.

top 2,000 ASINs in the Toys category with the highest glance view count and extract the forecast distribution and observed demand information only for those specific ASINs. The results should only be for forecasts created on 2016-11-20 and for each target week between 2016-11-20 and 2017-01-22.

columns: ASIN, Forecast Creation Date, Target Week, Model Name, Observed Demand, P50 Forecast, P60 Forecast, P80 Forecast, and P90 Forecast.

the data is stored in three tables. The first, weekly\_forecasts table, contains information about forecasts at the item level, the second, weekly\_demand table, contains information about demand at the item level (storing only records with non-zero demand values), and the third, asin\_details table, contains contextual information about the item.

SELECT TOP 2000 \*

FROM

(

SELECT fd.\*

FROM

(

SELECT f.asin, f.forecast\_creation\_date, f.target\_date\_begin, f.model\_name, d.demand\_unit\_count, f.fcst\_quantile\_50, f.fcst\_quantile\_60, f.fcst\_quantile\_80, f.fcst\_quantile\_90

FROM weekly\_forecasts as f

LEFT JOIN weekly\_demand as d

ON f.target\_date\_begin = d.demand\_week\_begin

WHERE f.target\_date\_begin BETWEEN '2016-11-20' AND '2017-01-22'

) as fd

LEFT JOIN asin\_details as a

On fd.asin = a.asin

WHERE a.category = “TOYS”

ORDER BY a.glance\_view\_count

)

;

Analyze the sample data file located at [https://goo.gl/VwZu82](https://amazon.hirevue.com/redirect/?url=https://goo.gl/VwZu82) which looks exactly like the output from the exercise in the previous question. Which of the two models A or B is, in your assessment, of better quality? Which metrics did you use to arrive at this conclusion and why? Your answer should contain the following:

• A short summary paragraph with your findings, methodology and recommendations

• Any insights and observations from the analysis you performed

• Assumptions you made, and potential limitations of your analysis

• Is there any extra data needed or questions you would ask to improve the analysis?

Also, please submit the code you used for the analysis in this part as an email attachment to [dmmozley@amazon.com.](mailto:dmmozley@amazon.com.)