# *Boozed'd Trees—Beer Sales Forecasting*

*Ludwig Schubert* [1], *Dan Zylberglejd* [2]

[1] ludwig@cs.stanford.edu
[2] dzylber@cs.stanford.edu

We predict sales volume of beer brands by unit size and region in Mexico during 2016 for brewery-conglomerate Anheuser-Busch Inbev. Augmentation with environmental data and surveying suited models lead us to an ensemble method combining LASSO and BAGGING with ABI's in-house predictions, over which we achieve a test error improvement in RMSE of 9.2%. We additionally present an optimal reallocation strategy when forecasts change after production has finished.

## The Challenge

Anheuser-Busch Inbev is the largest beer company worldwide, controlling main breweries such as Interbrew in Belgium and Anheuser-Busch in the US. Our project is focused on Grupo Modelo, its Mexican branch that produces, among others, the brands Corona, Modelo and Victoria. Mexico is a substantial beer consumer, and only in 2015 it obtained an income of around US$ 2.5bi from the brands analyzed on the scope of this project. After the acquisition of SABMiller, a big challenge that was posed by its new CEO is to improve accuracy of two month leading sales forecasts. There is not much historical data on past forecasts, but an internal analysis indicates that the consolidated monthly percentage error for Mexico is around 15–20%. To reduce unnecessary waste and ensure they meet customer demand, Modelo needs to decrease this number, and we implement a Machine Learning approach to tackle this great challenge. It can generate a considerable impact not only from a profit perspective, but also as a step to push a data-driven culture to its business.


Figure 1: Anheuser-Busch Inbev logo

## The Data

For each trio of UEN (region), unit size (e.g.: bottles, cans, barrels), and brand, ABI provides us with monthly sales in hectolitres since Jan/2013, as well as discounts and other pricing measures. For each example, we use past monthly sales up to one year.

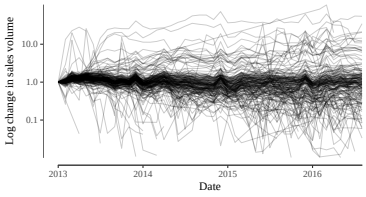


Figure 2: Monthly change of sales volume per trio UEN-Brand-Size on a log scale

We also scrape the web in order to augment our dataset by including temperature, holidays, soccer calendars and economic inputs (inflation, GDP). We carefully align the data with the information we would have had "at the moment of training". In other words, since we are predicting 2 months ahead, some features need to have a 2-month delay (such as past sales), while others don't, since they are known in advance (holidays, soccer calendar).

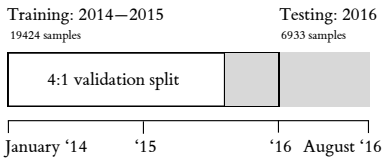


Figure 3: Training-Validation-Test split

Finally, we perform feature engineering on the main features. We

incorporate the difference between consecutive month sales, and also group past sales by UEN, unit size, brand, and also all the pairwise combinations (6 in total).

THE FINAL DATASET then consists on 25694 examples with 149 features and 1 numerical output. We split it into train, validation and test set following chronological order. We run all of our experiments on the training set followed by eventual checks on the validation set. The test set was only used for the final model assessment.

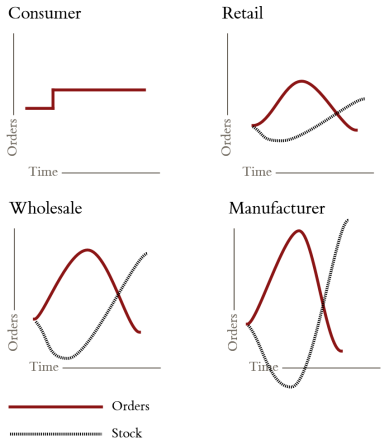


Figure 4: Illustration of the bullwhip effect and the outsized distortions as we go through the supply chain

## The Metric

Industrial engineering and supply-chain management literature mentions the so called BULLWHIP EFFECT [Chatfield et al., 2004]. This phenomenon occurs as inaccurate forecasts propagate backwards through a supply chain, leading to inefficiencies in inventory management that grow superlinearly on the size of the error. To account for this effect, we agreed with ABI to use a metric that penalizes errors quadratically for performance evaluation. We pick Root Mean Squared Error (RMSE).

WE ALSO report other types of metrics which ABI executives are more used to. However, it is important to reinforce that from a profit-oriented perspective RMSE is the most important evaluation criterion to look at. In particular their use of Mean Average Percentage Error (MAPE) could be misleading: it grows linearly in terms of the numerator, and the denominator inflates errors for small sales, which goes in the opposite direction of maximizing profit.

$$\text{RMSE:} \quad \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{Y}_i - Y_i)^2} \qquad \text{MAPE:} \quad \frac{1}{n}\sum_{i=1}^{n}\left|\frac{\hat{Y}_i - Y_i}{Y_i}\right|$$

## The Model

We start our modelling with a baseline, using CONSTANT YOY (year over year), which predicts the same volume for each trio (UEN-unit size-brand) as the previous year.

WE THEN JUMP INTO linear models, applying LINEAR REGRESSION and then adding an L-1 penalty to its coefficients to obtain a LASSO model. It is worth mentioning that our linear regression had high variance due to multicollinearity (especially economic inputs that changed only 3 times per year and were highly correlated - GDP, % change in GDP, Touristic GDP). LASSO took care of it, and produced results with low variance and similar training and validation errors. However, there was a large bias since the training error was still high.

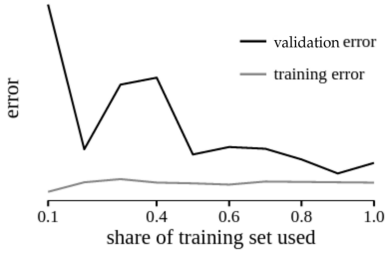


Figure 5: Convergence of training and validation error when the full training data set is used. (Linear Regression)

To DECREASE OUR BIAS, we explore more flexible models such as tree-based ones (given its success in other applications such Kaggle competitions [Kaggle, 2015] [Kaggle, 2014]). For BOOSTING, we tuned the max-depth of each tree and the number of trees using 5-fold cross validation (the step size we set to 0.01). Surprisingly for us, it does not perform well on the validation set. We applied RANDOM FORESTS with the main hyperparameter (number of possible variables on each split) set to its default, with some success. We then tuned it using out-of-bag error (faster than Cross Validation) and, surprisingly, found that instead of having a subset of possible features for each split, it is optimal to allow for split in any single feature.

This means we are not introducing any randomness on each split and, thus, implementing BAGGING, which led to our best results so far. We believe that the common reason for which BOOSTING and RANDOM FOREST did not perform well is the high order interaction effect of our data. Since each trio behaves differently from other trios, we believe we have a strong interaction effect of order at least 3. For high order interaction effects, BOOSTING is not suited well if we are using simple base learners with low depth, and the limitation of which variables we can pick at each split on RANDOM FOREST does not allow to fully capture those interactions either.

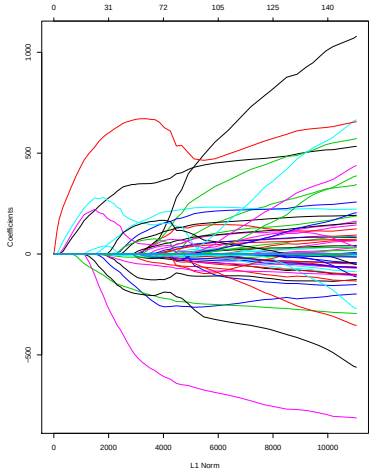


Figure 6: LASSO coefficients path. The largest coefficients are associated to brands and months.

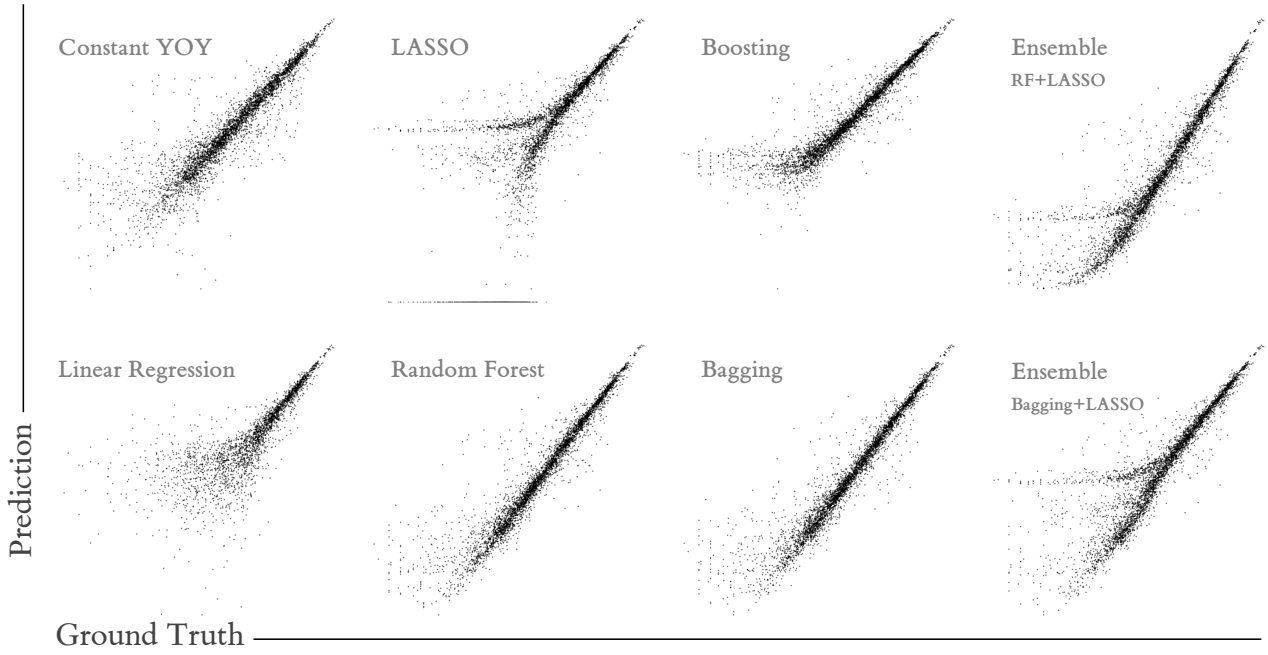


Figure 7: Predicted over actual Hectolitres for the validation set on a log–log plot. A perfect prediction would display as a 45° line. Note that we penalize with the squared of the error, so good predictors should be sharp on the top right. LASSO performs well even though its predictions spread out in the bottom left corner as those values are the least relevant for the resulting RMSE.

Given that we obtained good performance with both LASSO and BAGGING, we decide to combine both. We opt to ensemble them in a naive way, by just taking the average of their predictions (which works as something like a uniform prior to the model weights, leading to a regularization effect). Since those functions define very dif-

ferent surfaces, we expect a good ensemble effect, with errors cancelling each other and less sensitivity to a particular model (i.e., we expect to decrease our overall bias as well as variance, respectively). We report the validation error for all the mentioned models, and pick the ensemble of LASSO with BAGGING as our final one. We are now ready to apply it to the test set.

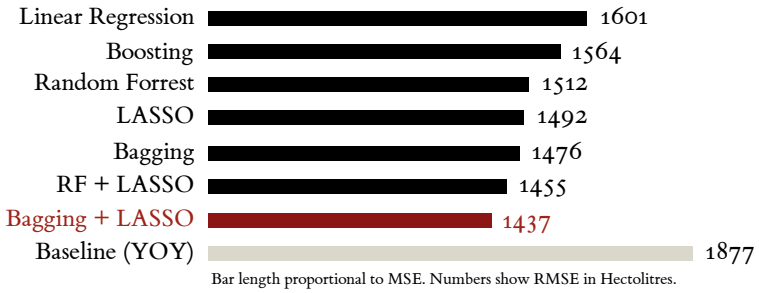


Figure 8: Performance of different models on the validation set.

## The Results

We run a complete backtest on the test set (Jan-Aug '16). More specifically, for each month that we are forecasting, we use all available data up to 2 months before it, retrain our model and make the predictions. We do it in a rolling window basis, so by the end of the backtest we have predictions for each month in the test set, exactly as if we have implemented our model to ABI's routines and made those forecasts by then. This allows us to fully trust our results and generate reliable estimates for future test errors (possibly even pessimistic, since we will naturally have more data in the future).

THE TABLE on the right describes our results quantitatively. Some important conclusions can be drawn:

1. The model that ABI is currently developing achieves an RMSE of 1883, which is believed to be already better than their current system. Our model achieves 1743, which is a great improvement over it. When we combine our model with ABI's through simple averaging, the resulting model achieves 1709, which corresponds to a 9.2% overall improvement. So we consider our approach successful, and recommend it to be implemented.

2. When analyzing other types of metrics, our model seems to be slightly inferior than ABI's model. It outperforms ABI on MAPE aggregated by UEN, and underperforms it if aggregated by brand or unit size. We once again restate that even though more interpretable, this measure is not representative of potential profit, and is also distorted by aggregations with low volume.

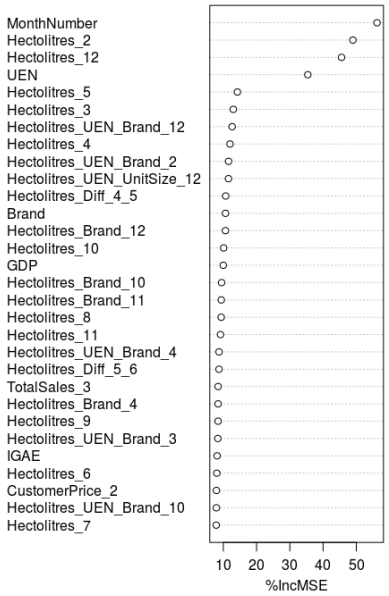


Figure 9: Most important variables for Bagging on the training set, measured by % improvement on total MSE. The most relevant ones are seasonality (month number), past sales, region, brand and economic conjuncture.

| Metric | ABI | Ours | Both |
|---|---|---|---|
| **RMSE** | **1883** | **1743** | **1709** |
| MAPE | 3.8% | 4.8% | 4.2% |
| " by UEN | 8.4% | 8.3% | 7.6% |
| " by Brand | 12.2% | 13.0% | 11.7% |
| " by Size | 8.0% | 9.8% | 8.3% |

Table 1: Performance on the test set for ABI's model, the model we developed (ensemble of Bagging and LASSO based on the mentioned features), and a model which consists on simply averaging our predictions with ABI's. We report results for both Root Mean Squared Error and also different aggregations of Mean Absolute Percentage Error.

3. Even though we don't recommend the MAPE as the optimal metric, we still achieve good results on the average monthly MAPE over the test set. We achieve less than 5% error, ABI's achieves less than 4% and the combined one around 4%.

4. Other advantages of the model are worth being mentioned: it deals well with missing data (the bagging part of the model uses surrogate splits for this purpose); each monthly model can be trained in a couple of hours on a standard local CPU; it is flexible to incorporate new types of data, such as weather/humidity (which were not used yet), or even analysts forecasts.

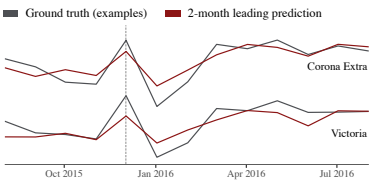


Figure 10: Actual sales (gray) and predictions (red) on the validation and test set (separated by the dashed line), for the two mostly consumed brands in Mexico: Corona Extra and Victoria.

## The Improvement

We extend the model to adjust itself dynamically. Predictions made with 1 month in advance tend to be more accurate than those made 2 months ahead. Since the production plan starts with 2 months in advance, ABI cannot change the amount of beer and bottles produced. However, they can reallocate it differently through the UENs. This can enhance logistics and even sales/marketing planning. So each month we train a model to predict sales one month in advance, and adjust our previous predictions under the constraint the the sum of the hectolitres for each pair of brand and unit size remains constant. We first analyzed it from a CSP perspective, but it turns out this optimization can be solved analytically through Lagrange multipliers (appendix).

Running a backtest on the same format as previously described, we obtain an improvement from an RMSE of 1743 to 1701 for our model. Curiously, ABI's model gets worse, increasing its RMSE from 1883 to 1966. The combination of both models also shows an improvement, going from 1709 to 1692. Thus, again, we recommend implementing the dynamic regional reallocation.

## The Next Steps

This project explored an application of Machine Learning to an industrial field such as food & beverages. It shows that a data-driven approach allows for enhancement of business performance, and that there is still room for other applications and improvement. Some possible extensions of our particular model, that came out of meetings with senior management of Grupo Modelo are:

1. Forecast exports and international markets.

2. Make longer term predictions, such as six months or a full year.

3. Predict sales for new brands or unit sizes that are yet unreleased.

## References

Dean C Chatfield, Jeon G Kim, Terry P Harrison, and Jack C Hayya. The bullwhip effect–impact of stochastic lead time, information quality, and information sharing: a simulation study. *Production and Operations Management*, 13(4):340–353, 2004.

Da Veiga et al. A comparison between the HoltWinters and ARIMA models. *WSEAS TRANSACTIONS*, 11:608–614, 2014.

Kaggle. Stores sales forecasting, February 2014. URL `https://www.kaggle.com/c/walmart-recruiting-store-sales-forecasting/data`.

Kaggle. Forecast sales using store, promotion, and competitor data, September 2015. URL `https://www.kaggle.com/c/rossmann-store-sales`.

## Appendix

Call $q_{old}^{(1)}, q_{old}^{(2)}, \ldots, q_{old}^{(n)}$ the old estimates for the quantity of beer to be sold for each pair UEN for a given pair (brand, unit size). Also, call $q_{new}^{(1)}, q_{new}^{(2)}, \ldots, q_{new}^{(n)}$ the new estimates for them, for the same pair (brand, unit size). Also, call $r^{(1)}, \ldots, r^{(n)}$ our final estimates. Our goal is to solve the following optimization problem:

$$\begin{aligned} \underset{r^{(1)} \ldots r^{(n)}}{\text{minimize}} \quad & f(r^{(1)} \ldots r^{(n)}) = \sum_{i=1}^{n} (r^{(i)} - q_{new}^{(i)})^2 \\ \text{subject to} \quad & g(r^{(1)} \ldots r^{(n)}) = \sum_{i=1}^{n} r^{(i)} - \sum_{i=1}^{n} q_{old}^{(i)} = 0 \end{aligned}$$

Calling $K_{new}$ the sum of total hectolitres produced by a given UEN and $K_{old}$ the sum of total hectolitres previously predicted by the 2-month in advance model, our constraint consists of $K_{new} - K_{old} = 0$. When we first designed this task, we thought about using Constraint Satisfaction Problem tools. However, it turns out that this optimization problem is convex and can be solved analytically using Lagrange multipliers. Define:

$$\mathcal{L}(r^{(1)} \ldots r^{(n)}, \lambda) = f(r^{(1)} \ldots r^{(n)}) - \lambda g(r^{(1)} \ldots r^{(n)})$$

From Lagrange multipliers, we need to set:

$$\begin{aligned} \nabla_{r^{(1)} \ldots r^{(n)}, \lambda} \mathcal{L}(r^{(1)} \ldots r^{(n)}, \lambda) = 0 \Rightarrow \\ r^{(1)} - q_{new}^{(1)} = \lambda \\ \ldots \\ r^{(n)} - q_{new}^{(n)} = \lambda \\ \sum_{i=1}^{n} r^{(i)} = K_{old} \end{aligned} \tag{1}$$

From the last equation in 1 we get that $K_{new} + n\lambda = K_{old} \Rightarrow \lambda = \frac{K_{old} - K_{new}}{n}$. Solving for an arbitrary $r^{(i)}$, we get the updated estimates:

$$r^{(i)} = q_{new}^{(i)} + \frac{K_{old} - K_{new}}{n}$$

This makes intuitive sense, since what it is doing is basically reallocating the estimated excess or shortage of beer/bottles equally over all the UENs while trying to stay as close as possible to the new, updated predictions.