Trick to enhance power of Regression model

BUSINESS ANALYTICS (HTTPS://WWW.ANALYTICSVIDHYA.COM/BLOG/CATEGORY/BUSINESS-ANALYTICS/) SAS (HTTPS://WWW.ANALYTICSVIDHYA.COM/BLOG/CATEGORY/SAS/)

SHARE (http://www.facebook.com/sharer.php?u=https://www.analyticsvidhya.com/blog/2013/10/trick-enhance-power-regression-model-2/&t=Trick%20to%20enhance%20power%20of%20Regression%20model) (https://twitter.com/home?status=Trick%20to%20enhance%20power%20of%20Regression%20model+https://www.analyticsvidhya.com/blog/2013/10/trick-enhance-power-regression-model-2/) (https://plus.google.com/share?url=https://www.analyticsvidhya.com/blog/2013/10/trick-enhance-power-regression-model-2/) (http://pinterest.com/pin/create/button/?url=https://www.analyticsvidhya.com/blog/2013/10/trick-enhance-power-regression-model-2/&media=https://www.analyticsvidhya.com/wp-content/uploads/2013/10/decision-tree1.png&description=Trick%20to%20enhance%20power%20of%20Regression%20model)

We, as analysts, specialize in optimization of already optimized processes. As the optimization gets finer, opportunity to make the process better gets thinner. One of the predictive modeling technique used frequently use is regression (Linear or Logistic). Another equally competing technique (typically considered as a challenger) is Decision tree.

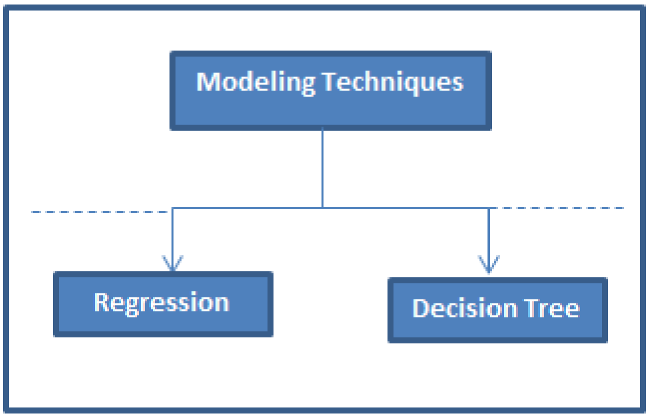
What if we could combine the benefits of both the techniques to create powerful predictive models?

(https://www.analyticsvidhya.com/blog/wp-content/uploads/2013/10/leap.jpg)

The trick mentioned in this article does exactly that and can help you improve model lifts by as high as 120%.

Overview of both the modeling techniques

Before getting into details of this trick, let me touch up briefly on pros and cons of the two mentioned techniques. If you want to read basics of predictive modeling, click here (https://www.analyticsvidhya.com/blog/2013/04/predictive-modeling-what-case-study-part-1/).



(https://www.analyticsvidhya.com/blog/wp-content/uploads/2013/10/modelling-org.png)

- **Regression** assumes continuous variable as is and generates a prediction through fitting curves for each combination input variables.
- **Decision tree** (CART/CHAID), on the other hand, converts these continuous variables into buckets and thereby segments the overall population.

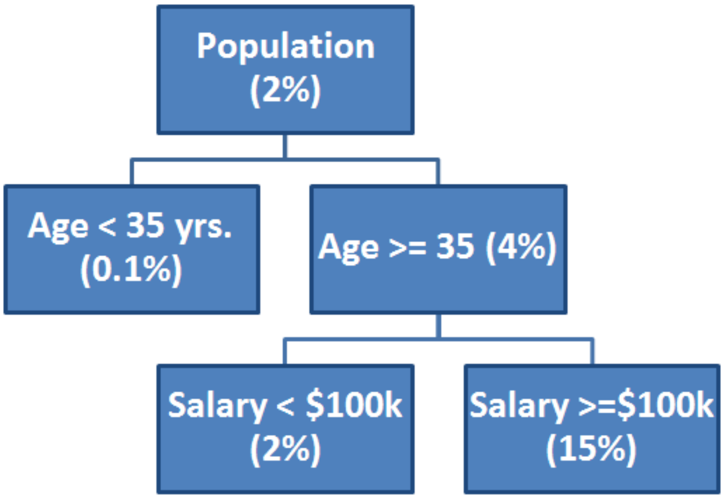
Converting variables into buckets might make the analysis simpler, but it makes the model lose some predictive power because of its indifference for data points lying in the same bucket.

A simple case study to understand pros and cons of the two techniques

If decision tree losses such an important trait, how come it has a predictive power similar to that of a regression model? It is because it captures the covariance term effectively, which makes a decision tree stronger. Say, we want to find probability of a person to buy a BMW.

Decision tree:

A decision tree simply segments the population in as discrete buckets as possible. This is how a typical decision tree would look like:



* Numbers in bracket are the response rate of that segment

(https://www.analyticsvidhya.com/blog/wp-content/uploads/2013/10/decision-tree1.png)

Even though the tree does not distinguish between Age 37 yrs. and 90 yrs. or salary $150k and $1M, the covariance between Age and Salary is making the decision tree's prediction powerful.

Regression model:

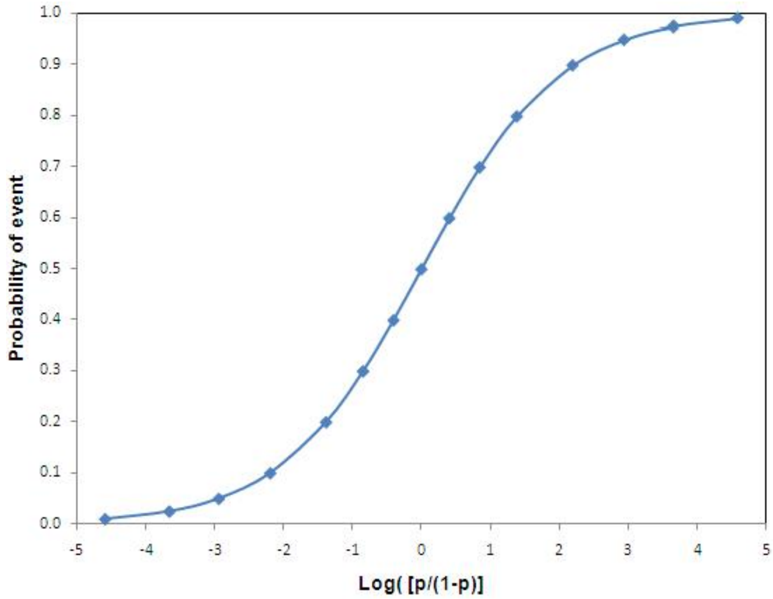
On the other hand, logistic regression makes use of Logit function (shape below) to create prediction. A typical equation would look like this:

$$Logit\ f = a * f_{Salary} + b * g_{Age} + c$$

(https://www.analyticsvidhya.com/blog/wp-content/uploads/2013/10/Equation.png)

where a, b and c are constants

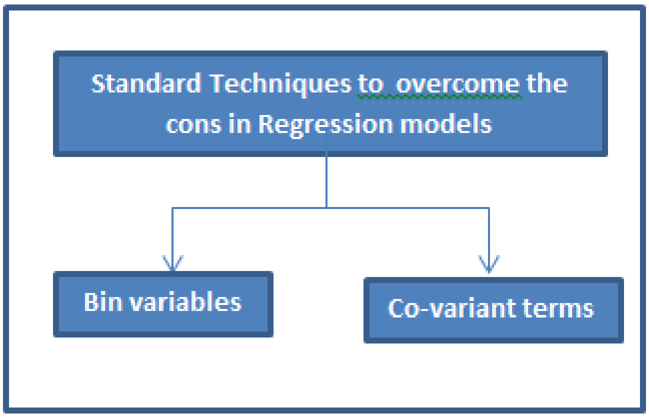
Shape of Logit function: (https://www.analyticsvidhya.com/blog/wp-content/uploads/2013/10/Formula-1.png)



(https://www.analyticsvidhya.com/blog/wp-content/uploads/2013/10/the_logistic_function.jpg)

Industry standard techniques to address shortcomings of regression modeling:

There are two basic techniques to capture covariance and discontinuity of the target variable:

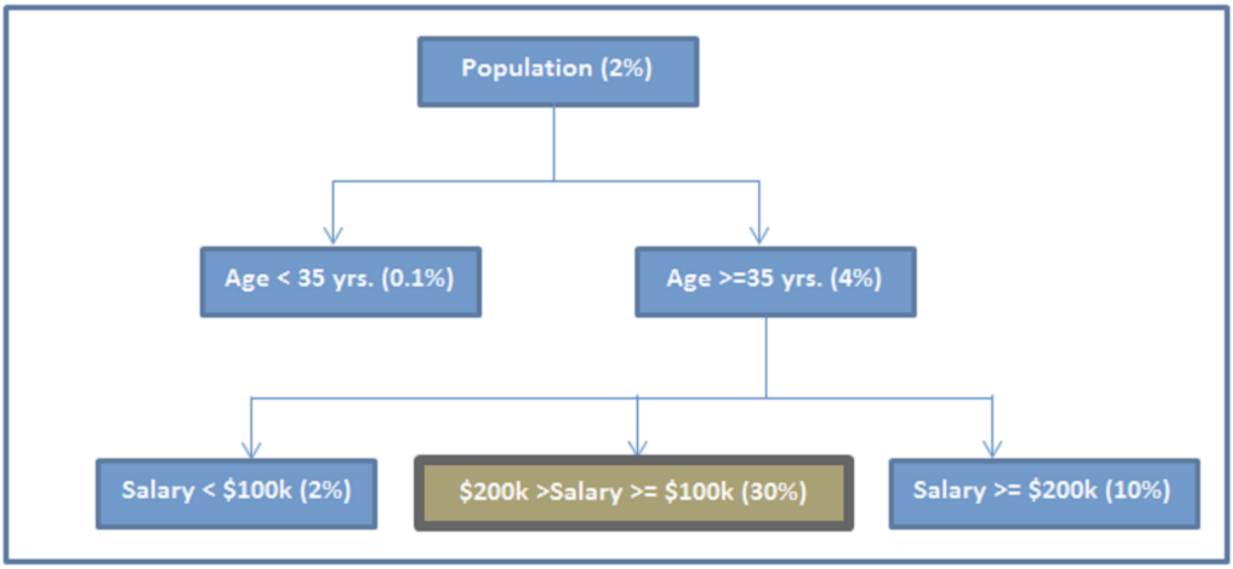

Standard Techniques to overcome the cons in Regression models
Bin variables
Co-variant terms

(https://www.analyticsvidhya.com/blog/wp-content/uploads/2013/10/Std-tech-org.png)

1. **Bin variable** with discontinuous relation: This is a technique used in almost all the models. If you are not familiar with this technique, it is nothing but flagging variables in the interval where a strong input variable shows a discontinuous relationship with the output variable.For example, 10% people break at least0 one traffic signal in US everyday. Only 3% of households with salary between $70k and $100k break traffic signal. Whereas, 11% of the rest of the population break traffic signal everyday, and this is almost uniformly distributed. In this case, to predict the propensity to break the signal, a good variable can be the salary bin $70k to $100k.
2. Introduce **covariance variables**: This is a technique used rarely. The reason being such variables are very difficult to comprehend and difficult to explain business.

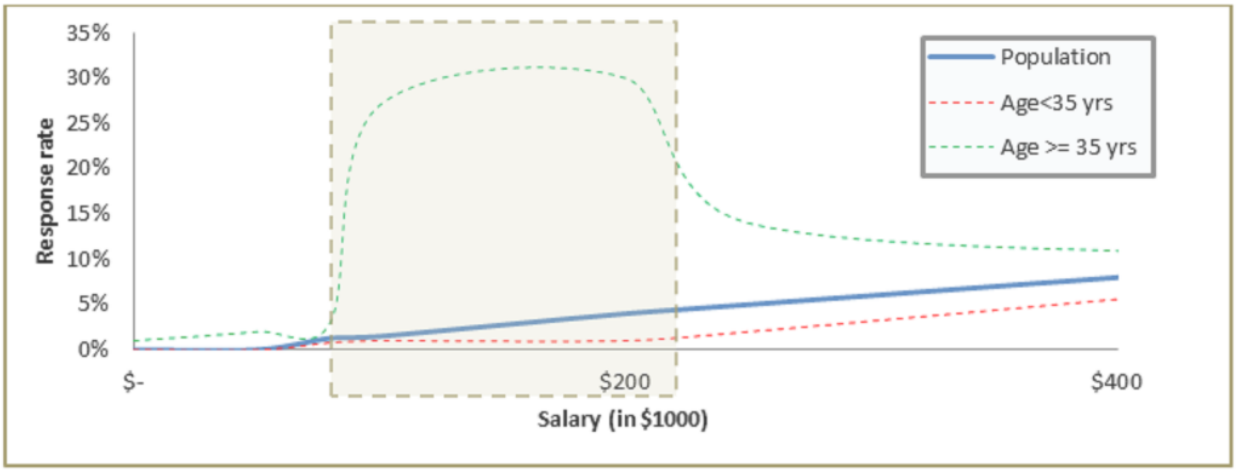
Each of these techniques capture co-variance and discontinuous variable well.

However, consider the following scenario where these approach have a high propensity to fail:



(https://www.analyticsvidhya.com/blog/wp-content/uploads/2013/10/DT-22.png)

Say in the population discussed in last section, people with salary between $100 K and $200 K and age above 35 years form a segment with exceptionally high BMW take up rate (30%). If we use the two discussed techniques, will the model capture this exceptionally high take-up segment? Will regression still be a better model compared to decision tree?



(https://www.analyticsvidhya.com/blog/wp-content/uploads/2013/10/graph1.png)

The answer is **NO** and **NO**, the regression will not be able to effectively capture this segment. Why does bin technique not capture this? The reason is that binning is done on a one-dimensional variable and in overall population salary bracket $100k to $200k might not even be different from rest. As you can see in the figure above, the response rate of the income bucket $100k to $200k is not differentiated when analyzed on overall population. But this bucket becomes very different when an initial cut of age >=35yrs is added.

Why does covariance term fail as well? The overall covariance between age and salary might not be significant for the overall population. Hence to have higher predictive power, the model needs an input that the trends of a particular segment are significantly different from rest of the population. Therefore,when the two problems i.e. discontinuity and covariance, exist simultaneously, regression model fails to capture the hidden segment. Decision tree, on the other hand works very well in these scenarios.

Have you guessed the trick?

Having worked on many of such problems, I find the following solution very handy. Both regression and decision tree have pros. ***Why not combine the pros of the two methods?*** I have used this technique in a number of models. And I was pleasantly surprised by the additional predictive power I got every time. There are two ways to combine the two methods:

1. **Introduce a new Covariant Variable:** A faster and an effective way to use the tool. Simply add this as one of the input variable for the logistic model.This Covariant term (a bi-variant bin) can be defined as:

$$\mathbf{Z} = \begin{cases} \textbf{1 if 200k > Salary > 100k \& Age >35} \\ \textbf{0 otherwise} \end{cases}$$

(https://www.analyticsvidhya.com/blog/wp-content/uploads/2013/10/eq3.png)

2. **Make two alternative models:** A time taking but more effective method in case the exceptional bin has reasonable size. In this method we build two regression models separately for the identified bin (Age > 35yrs. and $200k > Salary > $100k) and the rest of population. And add the two function by following logic.

$$H(x) = (1 - z) * f_x + z * g_x$$

(https://www.analyticsvidhya.com/blog/wp-content/uploads/2013/10/eq2.png)

Here, g(x) is the equation for the identified bin and f(x) is the equation for rest of the population. Z is same as defined in the last block.

I am sure you are wondering "What makes the lift go even higher than the lifts you saw in the two models?" If you are not I will really like to know your opinions on the reason.

How does it work? How do this trick create such impactful results?

Now let's try to think about what did we just do? Decision tree was coming out to be a better model because of a hidden pocket, which was two-dimensional bin. Even with the limitation of not using the continuous behavior of interval variable decision tree became very efficient to reduce false positive in a particular segment. By introducing the flag of this segment in logistic regression we have given the regression the additional dimension decision tree was able to capture. Hence by additionally using the continuous behavior of interval variables such as age, salary the new logistic regression becomes stronger than the decision tree.

Constraints of this trick:

There are two major constraints of using this technique,

1. Multi-collinearity: For models, where the VIF factor becomes unacceptable, the number of variables used to create the new input function should be reduced.
2. High Covariance: When overall covariance between two terms is high, this technique simply fails. This is because we will have to create too many buckets and , therefore, too many variables to be introduced in the regression model. This will introduce very high collinearity to the regression.

In general, I follow a thumb rule of not making more than 6 leaves in the parent tree. This, first of all, captures the most important co-variant buckets and does not introduce the two mentioned problems. Also, make sure the final bucket makes sense with business and is not merely a noise.

Final notes:

This trick helped me make my lifts rise by as high as 120% of the original lift. The best part of this trick is that it gives you a good starting point for your regression where you start with a couple of already proved significant variables.

What do think of this technique? Do you think this provides solution to any problem you face? Are there any other techniques you use to improve performance of your models (prediction or stability)? Do let us know your thoughts in comments below.