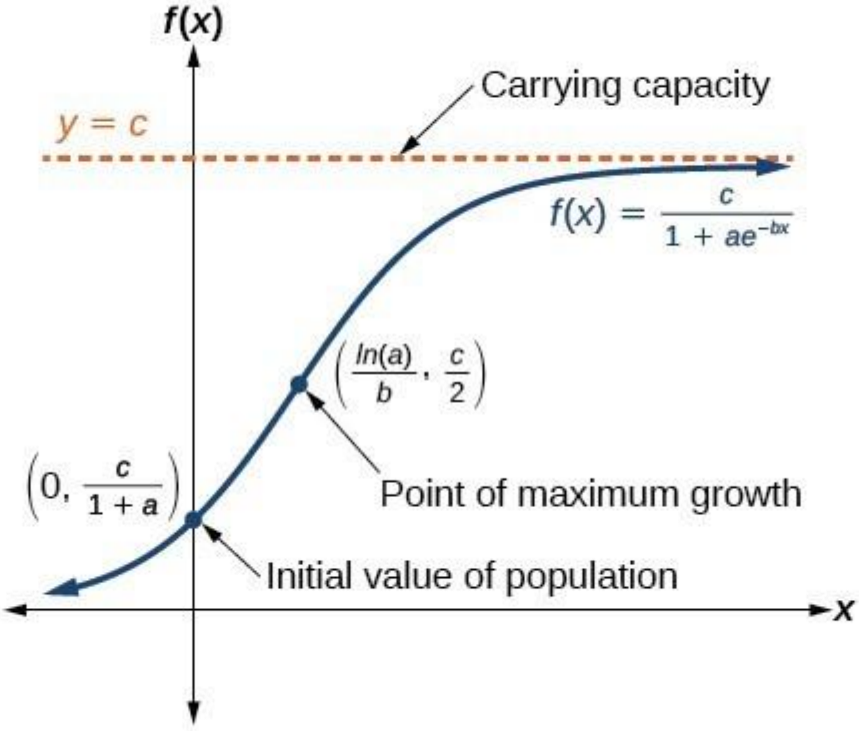
Andrew Rusinko III, Ph.D.
McKesson Data Science Challenge
May 23, 2017

The challenge was to predict the conversion rate of a generic entering the market. Data was provided that contained the first purchase date and customer commit date amongst other fields. These were collected into cohorts aggregated by purchase date (i.e., Cohorts1-31) and the number of conversions by date. However, the initial raw data file had to be sorted by purchase then commit dates to facilitate future processing (DataScienceTakeHomev2DataSetSorted.csv). All python scripts were run on Linux Mint using Python 2.7.11 (Anaconda).

McKesson.py converts the data into the form I chose to model. The processed data was echoed back to the console in CSV format or stored in a sqlite database. What I was most interested in was the cohort, the tracking date (date from the initialization of the cohort) and the cumulative conversion rate (per cohort, per date). This permitted me to create models for each cohort.

Two statistics to monitor would be the initial rate and the point of maximum growth (equivalent to an EC_{50}).


$f(x)$
Carrying capacity
$y = c$
$f(x) = \frac{c}{1 + ae^{-bx}}$
$\left(\frac{ln(a)}{b}, \frac{c}{2}\right)$
Point of maximum growth
$\left(0, \frac{c}{1+a}\right)$
Initial value of population
x

I used a logistic growth model for this exercise. We are most interested in accurately determining **c** (conversion rate per cohort). In this case, x was labelled “Track.” It is the number of days since the cohort began or Day#0, Day#1, Day#2…. etc. Y was the cumulative conversion rate for that cohort at Track.

Track = 0 or Day#0

I initialized the cohort and calculated an initial value of **a** based on a guess at the conversion rate **c**.

$$\mathbf{a} = \mathbf{c} / y_0 - 1$$

Track = 1 or Day#1

I calculated an initial value of **b** based on constant **a**.

$$\mathbf{c} = (1 + \mathbf{a}) * y_0$$

$$\mathbf{c} = (1 + \mathbf{a}e^{-\mathbf{b}}) * y_1$$

Solving for **b**:

$$\mathbf{b} = -\ln ((y_0/y_1 + y_0/y_1\mathbf{a} - 1) / \mathbf{a})$$

And then **c** was recalculated using constant **a** and **b**.

Track = 2 or Day#2

Fsolve (scipy curvefit) was used to estimate the values of **b** and **a**. Then **c** was computed from the y value on Day#2.

$$\mathbf{c} = (1 + \mathbf{a}e^{-2\mathbf{b}}) * y_2$$

Track = 3+ or ALL remaining days in study

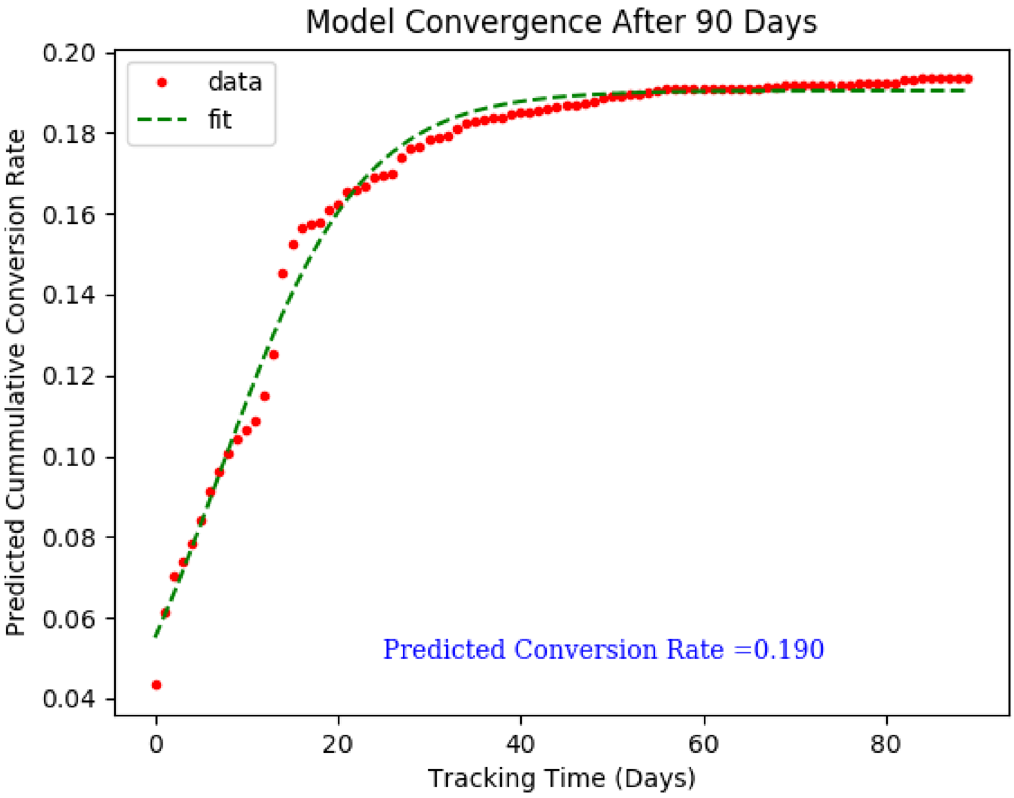
Fsolve was used to estimate the values of **a**,**b,c** using constrained optimization of a and b.

For each day, the mean of the conversion rate per cohort was also computed.

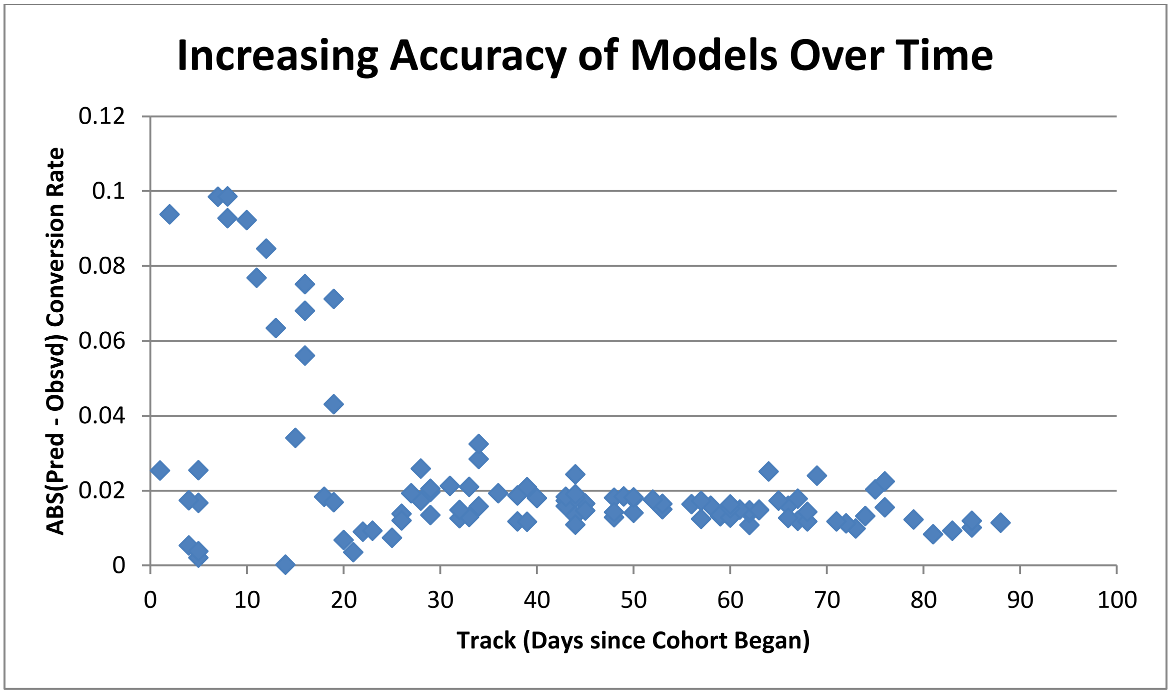
```
NewDate: 05/01/2013
Cohort1 0 0.25 0.0 0
Average Conversion Rate =  0.25

NewDate: 05/02/2013
Cohort1 1 0.293349388662 4
Cohort2 0 0.25 0.0 0
Average Conversion Rate =  0.271674694331

NewDate: 05/03/2013
Cohort1 2 0.286738076198 5
Cohort2 1 0.295705279748 4
Cohort3 0 0.25 0.0 0
```

```
Average Conversion Rate =  0.277481118649

NewDate: 05/04/2013
Cohort1 3 0.274020866258 7
Cohort2 2 0.28638803504 7
Cohort3 1 0.292415859843 5
Cohort4 0 0.25 0.0 0
Average Conversion Rate =  0.275706190285

NewDate: 05/05/2013
Cohort1 4 0.247290622687 8
Cohort2 3 0.26642763016 8
Cohort3 2 0.289813389882 6
Cohort4 1 0.296820809249 5
Cohort5 0 0.25 0.0 0
Average Conversion Rate =  0.270070490396
.
.etc.
```

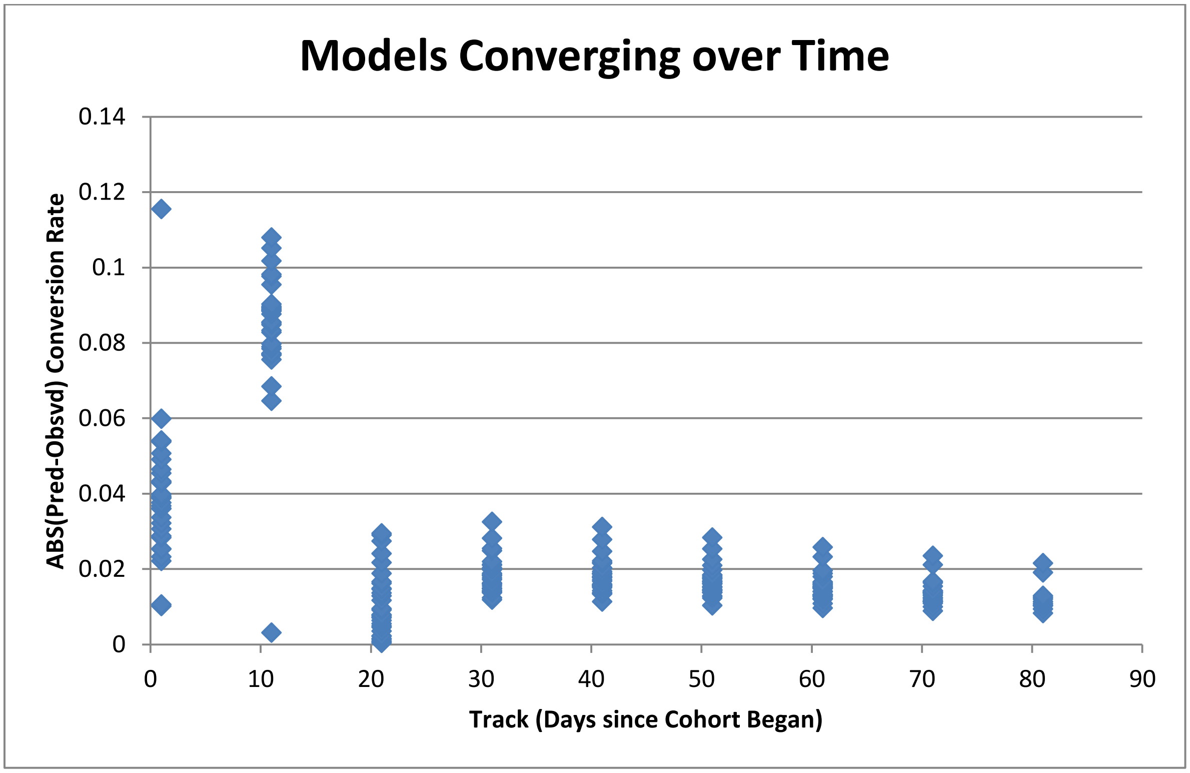
I was able to get a model for each cohort that became more accurate as it matured. Cohort#1 is shown below.



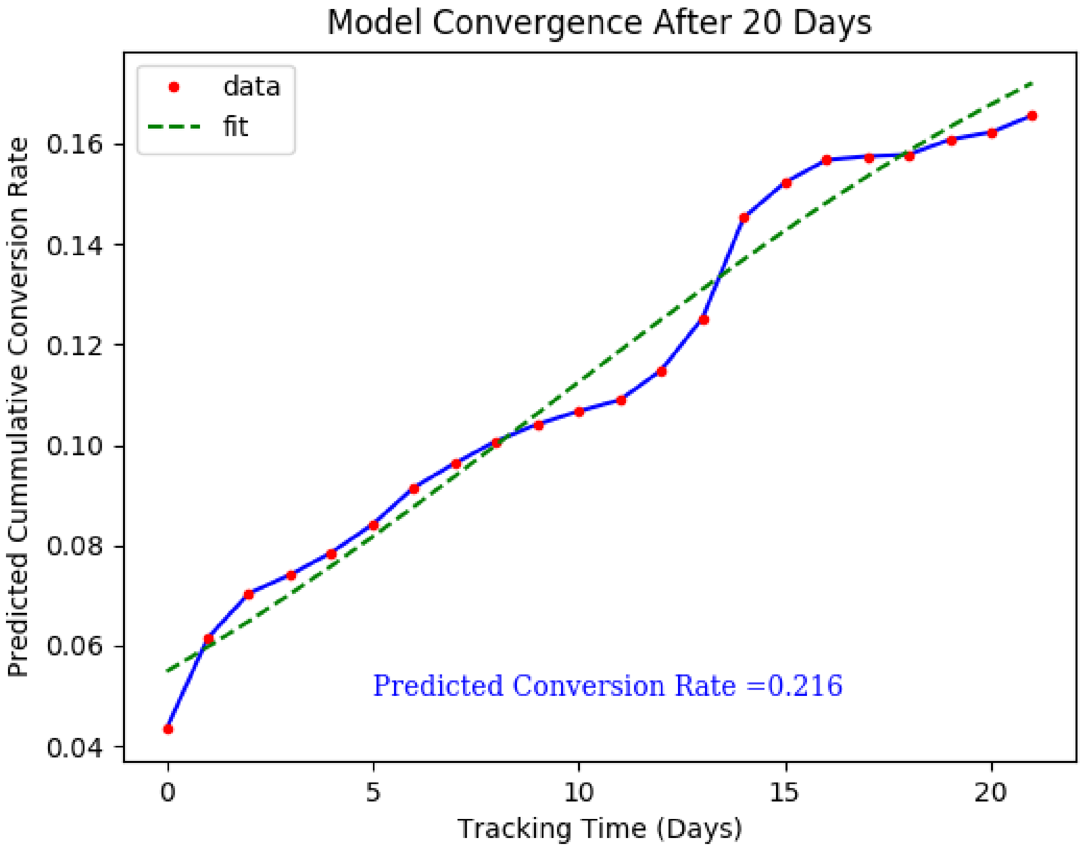
The predicted Conversion Rate was 0.190 compared to the observed value of 0.199.



Random sampling of calculations versus actual values demonstrate that the difference between observed and predicted values decrease as the models mature.



Periodic sampling of calculations versus actual values also demonstrate that the difference between observed and predicted values decrease as the models mature.



Other peculiarities were noted as well. It looks like there might have been something else going on to boost sales after 12 days. A more advanced model could be constructed to take this into consideration.