SOCIETY OF ACTUARIES

Article from:

Forecasting & Futurism

December 2014 – Issue 10

A Nearest Neighbors Approach To Risk Adjustment

By Geof Hileman and Claire Bobst

In the post-Affordable Care Act environment, health carriers are very limited in how much premiums can vary based on the risk levels of people seeking their insurance. In the absence of any other policy changes, this limitation would have incentivized insurers to avoid high-risk enrollees in favor of enrollees with fewer health conditions. This is clearly not a goal of the ACA. To minimize this risk, a risk adjustment program has been implemented to shift funds among insurers based on the relative risk of the people actually enrolled in their plans.

Risk adjustment models assign weights to various demographic and health-related categories according to the relative influence each factor has on an individual's cost. The weights corresponding to a given person's characteristics are then added up to determine their risk score. This risk score is normalized, meaning that a score of 1.0 indicates average health/risk, a score greater than 1.0 indicates worse-than-average health/higher risk, and less than 1.0 indicates greater-than-average health/less risk. These individual scores are then pooled to determine the average risk for a given group of people enrolled in a plan.

Behind the scenes, most risk adjustment models' weights are determined by regressions run on vast amounts of historical claims data. The method is quite effective and well-established techniques exist for its implementation, though it is still far from a "perfect" solution. The relatively low predictive power of these models is well-documented and, while this is mostly due to the variable nature of health care expenditure data, an opportunity potentially exists for a more powerful approach. By the nature of regression, all individuals in the sample are considered at once, and the incremental contribution of each of their characteristics to the average cost determines the outputted weights. Outliers, both high and low cost, are brought toward the mean, so that cost predictions for high-cost people tend to be too low and those for low-cost people too high, losing essential variation in the data.

A 55-year-old woman with multiple chronic illnesses, such as diabetes and heart disease, is unlikely to have similar costs to a 20-year-old male with no diagnoses. In regression risk adjustment models, they will, in however slight a way, have an influence on each other's predicted costs. To avoid this, what if, instead of running a regression on an entire dataset, we looked only at those people that most closely resembled the individual whose cost we were trying to predict:the people most similar in age, sex, diagnoses, prescriptions? In this way, we would only be considering the subset of the data most relevant to the individual of interest, and therefore the subset most likely to provide an accurate cost prediction for the individual.

A well-established algorithm, called *k*-Nearest Neighbors, can be applied to do just this. It consists of three simple steps:

1. Calculate the "distance" from the new data point to be classified to all the data points in the test set (note: dependent variable values are known for all points in the test set).

2. Determine the *k* data points with the shortest distances from the point in question. These are the "neighbors."

3. Average the dependent variable values of these *k* neighbors, weighting closer data points more heavily than those further away. This average is the approximated value for your new data point—the risk score.

The *k*-Nearest Neighbors algorithm is widely used in a variety of industries due to its simplicity, intuitiveness, and applicability to a variety of problems. Among these applications are the classification of breast tissue samples as malignant or benign based on a data set of known samples, the use of past weather data to create a stochastic weather generator and predict future weather in an area, or even audio fingerprinting, e.g., determining the identity of a song by comparing a short sample to a huge database of known samples (k=1 in this case). Nearest Neighbors is potentially applicable in most any situation in which past experience can be used to classify a new object.

Though the algorithm may seem simple and easy to implement, it is deceptively complex. In our application, we would like to predict the cost of an individual based on a set of people with known costs. Before we can do this, two main issues must be addressed. First, how do we determine the distance between two people? Our points aren't of the Cartesian (x,y) variety; instead, they are more complicated, consisting of a set of many different variables. We need to know how each relative difference and similarity in these variables impacts the difference in cost between two people, and therefore the "distance" between them.

The second issue is determining the optimal number of neighbors, *k*. The ideal *k* will minimize the error between the cost calculated by the algorithm and actual cost. *k* can be thought of as a smoothing parameter: it has to be large enough to smooth noise in the data but small enough to give an accurate estimate. A *k* value too small will be affected by noise, but a *k* too large takes into account irrelevant data points (at its limit, *k* is equal to the number of individuals in the sample and thus each individual is assigned the average cost, with closer neighbors weighted more heavily). Essentially, our choice of *k* has a tremendous impact on the accuracy of the Nearest Neighbors approach.

These two issues make apparent the work necessary to create a full-blown implementation of this algorithm. As such, our work has been of the proof of concept variety—investigating the idea to see if it has potential as an alternative approach to risk adjustment. To do this, we have been using R, a free statistical programming language, convenient in that it is both easy to use and provides open access to a huge number of packages written by programmers around the world.

We began our work in R by writing a script that would attempt to determine an effective distance formula. The idea here was actually to use a regression model, but with a subtle yet important difference from risk adjustment regressions: the model would return weights indicating the relative importance of each *difference* between two people in determining their *difference* in cost (how "far apart" they were).

THE K-NEAREST NEIGHBORS ALGORITHM IS WIDELY USED IN A VARIETY OF INDUSTRIES DUE TO ITS SIMPLICITY, INTUITIVENESS, AND APPLICABILITY TO A VARIETY OF PROBLEMS.

Unfortunately, this requires comparisons between every pair of people in a data file, a number that grows quickly with the size of the data. We realized that large amounts of computational power would be necessary to implement this regression, and that the results we were getting were unusable simply because we couldn't take enough comparisons into consideration. To temporarily deal with this problem we have been using weights from an already-established risk adjustment model as the distance formula coefficients. This is definitely not a perfect solution, and could affect the credibility of our results, but it at the very least provides us with a functioning distance formula for a proof of concept demonstration.

We then wrote a script in R to implement the *k*-Nearest Neighbors algorithm, and have been running various tests to look at how the results compare to that of a regression risk adjustment model. We have been using a data file containing 5,000 people, due to the fact that it takes about a minute for the NN algorithm to compare a given person to all 5,000 people in the file. Though a relatively small number of people to use, we've limited the sample to this size for now to avoid an even longer running time.

The program selects a given number of random people to classify from the data file using a specified starting seed. It then runs the Nearest Neighbors algorithm for a specified number of neighbors *k,* returning both the Nearest Neighbors error (NN cost – actual cost) and the regression error (regression cost – actual cost) for each random person.

CONTINUED ON **PAGE 28**

A NEAREST NEIGHBORS ... | FROM PAGE 27

The algorithm was run for 40 random people in our data file for various numbers of neighbors *k*. The mean absolute error using nearest neighbors for these 40 people was calculated for each *k,* as was the percent of cases where nearest neighbors produced less absolute error in predicting cost than the regression model. No "best" value for *k* emerged, though more tests should be done to reduce the effect of data variability. All that is clear is that the number of neighbors should be greater than one and less than 50, which makes sense given the earlier discussion of *k*.

Error for 40 Random People at Various *k* Values

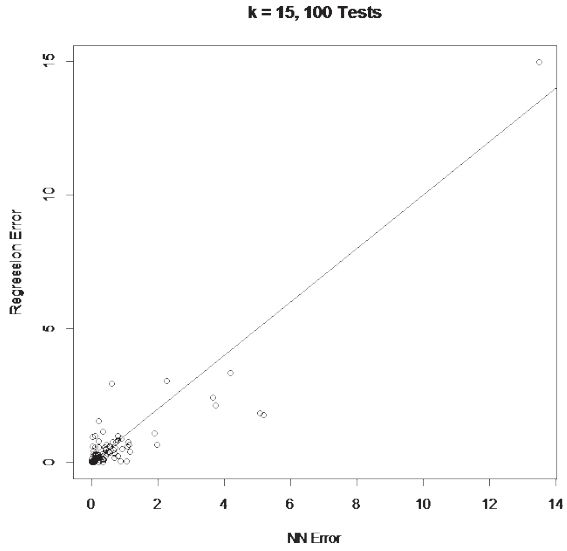
Total Number of Tests = 40		
k (number of neighbors)	NN mean absolute error	Percent of cases where NN abs. error is less than regression abs. error
1	0.9096	42.5%
5	0.7614	32.5%
10	0.7633	42.5%
15	0.7848	32.5%
20	0.7311	45.0%
50	0.8410	40.0%
Regression	0.6947	

Choosing an arbitrary value of *k*=15, 100 random people were generated and their cost predicted by the algorithm. The results indicated very similar absolute error for Nearest Neighbors and for the regression model, with NN producing less error for almost half of the people.

Error for 100 Random People, *k*=15

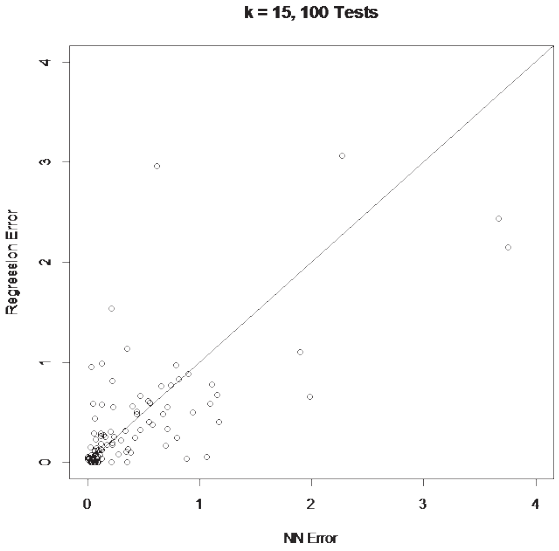
NN mean absolute error	0.70
Regression mean absolute error	0.61
Percent of cases where NN abs. error is less than regression abs. error	48%

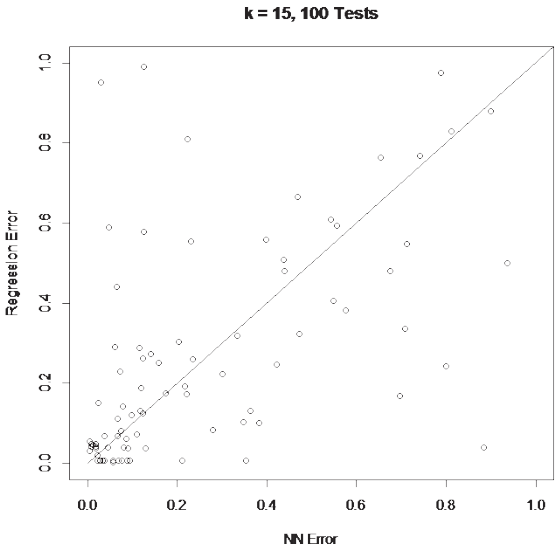
THE MAIN ISSUE WITH A NEAREST NEIGHBOR APPROACH ... IS THE COMPUTATIONAL COMPLEXITY OF ITS IMPLEMENTATION.



The above plot displays Nearest Neighbors error versus regression error for each of these 100 randomly chosen individuals (one point indicates the results for one specific individual). One clear outlier is apparent, with large error produced by both methods. The vast majority of the data points, however, cluster around the unit square, providing a visual representation of the typical accuracy of the two models. The line y = x allows us to further compare the two: data points above the line indicate individuals for which the regression model produced more absolute error, while for points below the line Nearest Neighbors produced more error. This line seems to generally segment the data, supporting the fact that 48 percent, or about half, of the cases had less error produced by NN than by the regression model. The below plots zoom in on the data, the first on the interval [0,4], then on the unit square, to examine the majority of the data more closely and further illustrate this point.

Further experimentation has indicated that, as is the case with regression-based risk adjustment models, almost all values of *k* produce cost estimates that are too low for high-cost outliers. This again makes sense, but could be improved by increasing the size of the data file and thereby increasing the probability of finding neighbors that are





similar and have similarly high cost. Even with a larger data file, a smaller *k* will be ideal for these outliers, but this increases the extent to which their cost estimates are impacted by variability in the data. It is possible that the regression approach will remain preferable for these high-cost and/or rare-diagnosis people. On the flip side, Nearest Neighbors can produce very accurate cost estimates (0.1 error or less) for zero-cost people. This can be done using essentially any number of neighbors, due to the large number of people with no diagnoses, and thus zero or very low cost, in a given data file (specifically, zero-cost people make up 12.3 percent of our data file of 5,000 people).

We can conclude with several ideas for creating an effective implementation of the Nearest Neighbors algorithm. All generally revolve around the use of sufficient computing power. First and foremost, it is necessary to determine an optimal distance function, as having such a function that will allow definitive conclusions regarding how NN compares to traditional regression models. This will require processing a very large amount of data, as again the number of comparisons between people grows exponentially with the size of data. Not all people need be compared, but more comparisons will lead to a more accurate distance formula. Going along with this idea, the size of the test set should ideally be increased in hopes of improving predictions for high-cost outliers as well as for average-cost people.

The main issue with a Nearest Neighbors approach, both to risk adjustment and to the various other fields in which it is used, is the computational complexity of its implementation. We have to compare each new person to be classified to every other person in the test set, which simply takes a very long time. As such, there is a tradeoff between execution time and error: more data means slower execution time and less error, less data means faster execution time and more error. It is a fundamental issue. Some faster, modified Nearest Neighbor algorithms do exist, and it seems that these take one of two approaches: reducing the size of the data set in some way, or using some sort of tree structure to divide the test data into groups with similar characteristics. For our data set, we could take the first approach by reducing the number of variables involved, possibly by grouping together similar diagnoses (i.e., diagnoses in the same category but with different levels of severity). We did experiment with a simple tree structure, specifically by dividing our data into pre-defined demographic groups. Either approach could decrease the computational issues inherent in the problem.

CONTINUED ON **PAGE 30**

A NEAREST NEIGHBORS … | FROM PAGE 29

k-Nearest Neighbors is a potential alternative to the traditional regression approach to risk adjustment. Despite being a very simple algorithm, there are layers of complexity underlying its implementation, many potential questions to be raised and problems to be addressed. While a Nearest-Neighbors approach may not ultimately prove ideal for many risk adjustment applications, our preliminary evaluative efforts have suggested that this approach has potential for improving the predictive accuracy of risk adjustment algorithms. ▼

Geof Hileman

Geof Hileman, FSA, MAAA, is director of actuarial studies with Kennell and Associates, Inc., in Raleigh, N.C. He can be reached at ghileman@kennellinc.com.

Claire Bobst

Claire Bobst interned with Kennell and Associates, Inc., in 2014, and is currently a senior mathematics major at the College of William and Mary.