**Personalized Composite Score Calculation**

In this document, I describe how I plan to calculate the “Personalized Composite Score” for the neighborhoods in San Francisco bay area.

The idea is to calculate a composite score for the zip codes in our dataset personalized by the user input.

The idea is to display a set of sliders for the user to set their importance level to various features we have collected data for, such as school rating, crime index, renter occupied housing percentage, poverty rate, number of parks, etc.

I plan to present the features for which we have data for (we do not want to go and collect more data at this time, I believe), to many of my friends, colleagues and ask them which of those they would consider when looking buying a house in a neighborhood.

Based in the feedback, I plan to include 8-10 features in this calculation.

Note, in the following calculation I am not using linear regression coefficients, the reason for that is explained in the other document I am uploading that talks about the overall presentation aspect of our project.

Steps:

* Dataset preparation – feature selection.
* Calculate normalized or scaled measure for positively and negatively correlated features.
* Combine the normalized data values for the zip codes with the weight the user wants to give to the various features and calculate the personalized composite score.

## Dataset Preparation – feature selection:

For this personalized composite score calculation, I plan to select the features as per the plan I outlined earlier.

Then from the merged dataset we have collected / prepared I would select the columns that correspond to those features only.

I pan to drop features/columns like the average median house price, the median household income, unemployment rate, etc.

This dataset will be used in the next preparation step as described below.

## Normalized or scaled measure for positively and negatively correlated features:

In the following discussion, a positively correlated feature would mean a feature that is considered favorable or contribute positively for a neighborhood. Examples of this type of feature would be school rating and number of parks in the neighborhood, where more is better.

And a negatively correlated feature would mean a feature that is considered unfavorable or considered to contribute negatively for a neighborhood. Examples of this type of feature would be crime-index, poverty rate, where more is worse.

For every feature / column, for every data value in that column I would do the normalization as follows:

M => Maximum measure in that column, m => Minimum measure in that column

For positively correlated feature:

Normalized measure = (Original measure - m) / (M – m) X 100

For negatively correlated feature:

Normalized measure = 100 - (Original measure - m) / (M – m) X 100

Note, that the normalized measure is always a number between 0 and 100, irrespective of what the original measure’s range was.

## Personalized Composite Score calculation based on user provided weight for the features:

User provided weights would be scaled between 0 and 1. If the UI (User Interface) lets users select weight values between 0 and 100, those will be rescaled to floating-point numbers between 0 and 1. For example, a user weight of 90 will be rescaled to 0.9 and 36 will be scaled to 0.36.

Now once the user has given weights to the various features, the personalized composite score (PSC) is calculated as:

Ui = User provided weight for ith feature, f= Number of features, Ni = Normalized / scaled measure for ith feature.

PSC = Ui X Ni / f