# Project Proposal

1. What your project is supposed to do.

The intent of this project is to build software that can help students using data science for categorization or simple time series analysis, using basic statistical model such as *k-nearest neighbor* and *linear regression*.

***Nice to have:*** Time allowing, we can move from static implementation of categorization algorithms to a plugin architecture. The user can write their own plugin to apply a different categorization method. This improves justification for the software, in terms of the question *why someone would want to* ***use*** *such a program*.

The proposed software will graph a two-dimensional dataset and create a line of categorization when available (with both the listed algorithms, this is the line of best fit). It will color each data point according to its category. Details about individual data points will be visible on hover.

On startup, appropriate random data will be generated. A model will be retained, using the provided data as the training set, and the user given the opportunity to then add custom data points as test cases.

A file specification will be provided along with a means to load custom data. Why anyone in their right mind would want to write a program to do what  
yours is going to do.

I am motivated to build this software because it intersects with my planned master’s capstone. I am preparing to take a data science course this summer so that I can add basic forecasting to my app, which I hope will be the major deliverable in my capstone.

2.a. Why someone would want to *use* such a program"

In general, anyone beginning with data science and also using Java could benefit from writing an app like this, while anyone just learning about data science would do better using standard tools like R. Nonetheless, students exploring data science benefit from a system that allows them to explore these models. Someone could find niche uses for it in practice as well, as it should be a light-weight app that can run different classes of algorithms on the data and immediately graph it with a standard view.

1. The form of the input and output, perhaps with a sample dialogue.



There is a graph in a Swing window that depicts the data as described above. The data visualization will show data points distributed across x and y vectors in a scale determined to fit the data. When available, the categorization slope will be rendered (this is algorithm dependent). Colors will be assigned to each group, and the data points are rendered in the appropriate color. Hovering over a data point will render its raw data in a tooltip.

The form of the input will be as indicated in that document, with each data point providing its pair of data as {x,y}. However, specifically, it makes sense to use JSON if I can use a parser like gson.

***Nice to have:*** Optional support for additional data to be rendered in the tooltip. This would follow the form {x, y, data}.

The main algorithm applied to the data will be the k-nearest neighbor algorithm as implemented in a standard library. My real tasks here are:

1. Create a plugin model that can be meaningfully applied to the current problem (data analysis) to extend the capabilities of the software.
2. Create or utilize a label-to-number algorithm to provide quantitive and unbiased data from strings in the case that some of the data is not numeric. For example, data about butterflies might be of the form { length in mm, dominant color } In such a case, each color must be given a number.
3. Create a UI and method that can be run against the data input and the analysis output to display the graph. Care must be taken to scale the graph to the data. A rough algorithm for this is:

results = categorize(method, data)

scalex = results.maxByKey(‘x’) /

frame.getSmallerDimension()

scaley = results.maxByKey(‘y’) /

frame.getSmallerDimension()

for(result: results) {

tooltip = getTooltip(result)

addPointToGraph(result, tooltip, graph,

scalex, scaley)

}

1. Create a formal JSON input file format specification for data input.

## Regards that capstone

I am interested in enabling learning experiences with technology and addressing the scheduling and motivation needs of students. I want to write a mobile app that is also a service. The app part would allow the user to configure which external apps should have preference. The service would allow external apps to forgo notifications, instead reporting performance metrics for individual tasks in the accomplishment of some goal, and optionally also specific notification messages. The metrics would be used to determine which apps should trigger notifications now.

This sort of service is best fit for educational apps and apps that help in the performance of a repetitive/maintenance task. Purely commercial apps (like games, or shopping apps), don't necessarily benefit by sharing the user's attention to help them best achieve their own goals.

The notifications would not blindly follow weak performance, nor simply function to preserve retention. Instead, they would be scheduled based on behavior, and in light of outcomes in future performance metrics on the same task. For example, if one such app is a dieting app, and you tend to go off your diet on Thursdays, at lunch, at McDonalds: At first, the app would blindly notify you of your performance right before you would go off your diet. But it might turn out that in fact if the notification comes an hour before, you perform better because you have time to come up with an alternate plan. Someone else might have better outcomes if it comes an hour or two later, almost guilt-tripping them into conformance.

We're trusting the app to help create the context that is best for independent learning, something that normally is largely the responsibility of the student. The app does not prescribe a general pattern, it observes the user and conforms its notifications to fit their lifestyle with an evidence-based approach. -- My own undergraduate degree intersects with this in its preparing students to develop web and mobile apps. From here I need to focus on other aspects: systems (service) programming, research, behavior, and education.

It could be that it isn't just on Thursdays that the dieter is going off their diet.. it is instead any time they happen to be near that McDonald’s. The software can look at location data and pair it with performance to predict the right times to notify you, even if it doesn't follow a strict schedule.

With A/B testing (some software enabled with certain strategies for notifications, others not), we could then even get data back about how useful the pattern is. It may be that in general there is statistical significance in using this for notifications. Or it could be that only certain populations benefit. That could lead, or not, to further research.