Broadly, the problem is to infer, from written text, information about the demographic profile of the writer. To create a minimum viable product (MVP), I will initially limit the text to be used to the text portion of Twitter statuses (tweets), and I will initially limit the demographic information to be inferred to gender. In addition to extensions to other demographic attributes and to text from other media, the specific Twitter context suggests possible extensions to using non-text information (e.g. emojis included in a tweet, visual information from Twitter profiles, etc.), but the scope of this project, as herein proposed, will be limited to text.

My imaginary client is a consulting firm (Imaginary Marketing Associates, LLC, or IMA for short) that helps its clients with targeting their marketing efforts. IMA will use inferred demographic information to target marketing efforts for products expected to be of interest to particular demographic profiles.

For a couple of reasons, IMA will be interested in how the predicted demographic information associated with a tweet changes over time: first, because their clients will be reacting in real time; second, because they will be developing profiles of individuals that will depend on how their text is classified at different points in time. For example, if the typical characteristics of male and female tweets change over time, it would add confidence if the same Twitter profile produced tweets at different points in time that were classified as the same gender using models sensitive to different conditions that apply at those different points in time.

For the initial MVP, I will rely on streaming data to be obtained from Twitter via their API. I already have a developer account with Twitter, which will enable me to access random streams of public tweets.

The expected approach will involve applying a machine learning model to text embeddings. For example, in a preliminary analysis, I use Google's Universal Sentence Encoder to produce embeddings and then apply a gradient boosted decision tree model to those embeddings to predict gender. (I can impute the "ground truth" for gender for some tweets from the display name of the user, using available gender imputation software, such as the Python gender-guesser package. For training and evaluation of the model, I will rely on tweets for which I can impute gender from the display name with reasonably high confidence.)

From a machine learning point of view, the initial problem (gender inference) is a supervised binary classification problem. Later extensions (applying it to different demographic information) could involve multi-class classification, or possibly regression.

Deliverables will include: (1) code for initial data preparation; (2) code for initial model fitting and evaluation; (3) code for model updating (to account for changes over time); (4) code for model application (inference), to produce both unconditional and time-conditional predictions; (5) a slide deck presentation describing the project; (6) a written report describing the project, including statistical analysis.