1. Starting with the featMat.csv dataset, we have 21,000+ obsv and 30+ columns.
   1. **Technically Correct Data (Col Descriptive Names):** From the featureString.txt file, we have the column names. You can manually add that to your data (featMat.csv) or figure out a Python function that would assign column names to a pandas data frame.
   2. **Technically Correct (Column Types):** The next step is to ensure that your data is technically correct. The read\_me features allow you to understand what the categorical variables that are encoded using numbers are.
   3. **Consistency of Data:** Maybe you can check the values for each of the numeric columns, possibly with either the pandas-profiling approach or the describe() from pandas.
2. The Next Part has to do with trying to replicate the analysis from the paper. For the purpose of our assignment, I wanted you to focus on the left figure in Slide 24.
   1. **Scrolling Classifiers:**
      1. Doc ID/ Sessions 1-3 (were reading Wiki articles sessions) – on Day 1 for each user
      2. Doc ID/ Session 6 (were reading Wiki articles) on Day 7/8 for each user
   2. For their box plot, they had 3 different rows. I want you to read the paper in more detail to understand the differences:
      1. Intra-session: for each user, you are trying to see whether you can build a classifier to detect how they perform within a given session (e.g., in session 1, can I build a classifier that would classify each user vs everyone else and you do that for each of the four sessions)
         1. **Binary Classification:** The one vs all – was their choice for the experimental setup. They chose that setup which put them in the binary classification world since their response is user of interest vs the rest of the group. **Things to Do:**
            1. **The approach used in the paper.** To replicate this, you will need to programmatically recode the user id variable. Easiest approach would either be a “for loop” or “def a custom function” that would do that for you.
            2. How to define the test set? For a given doc ID, random sampling is fine. Technically, you would want to filter the doc id prior to running the setup function. **Because I am trying to simplify your life, run it for one session/doc id.**
         2. **Multi Class Classification:** where they would try to correctly classify each of their 40 users based on the data (y = u1, u2, …, u40).
         3. **Anomaly Detection/Single-Class Classifier:** Train the model to categorize each user and then see if new users are identified as the user you trained the model for
      2. Inter-session – you would want to test whether across sessions 1,2,3 and 6 can you model a user’s behavior. There are two approaches to test for this; I believe they used the simpler approach of mixing all the data from all the sessions and then running the experiment as we see in step 1.
      3. Inter-week study – train based on sessions 1-3 and test data based on session 6. This is NOT a random sample and requires a manual test and training set in pycaret.