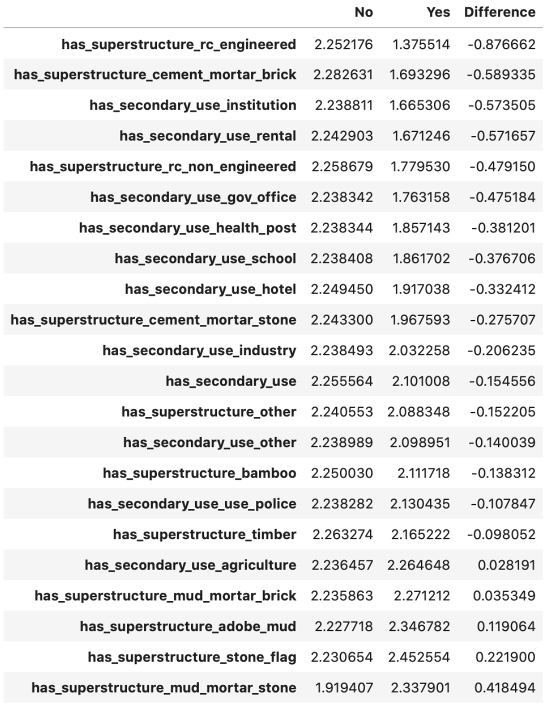
Presentation Script
Order:

1) Tamara
2) Jack
3) Emily
4) Marva



	No	Yes	Difference
has_superstructure_rc_engineered	2.252176	1.375514	-0.876662
has_superstructure_cement_mortar_brick	2.282631	1.693296	-0.589335
has_secondary_use_institution	2.238811	1.665306	-0.573505
has_secondary_use_rental	2.242903	1.671246	-0.571657
has_superstructure_rc_non_engineered	2.258679	1.779530	-0.479150
has_secondary_use_gov_office	2.238342	1.763158	-0.475184
has_secondary_use_health_post	2.238344	1.857143	-0.381201
has_secondary_use_school	2.238408	1.861702	-0.376706
has_secondary_use_hotel	2.249450	1.917038	-0.332412
has_superstructure_cement_mortar_stone	2.243300	1.967593	-0.275707
has_secondary_use_industry	2.238493	2.032258	-0.206235
has_secondary_use	2.255564	2.101008	-0.154556
has_superstructure_other	2.240553	2.088348	-0.152205
has_secondary_use_other	2.238989	2.098951	-0.140039
has_superstructure_bamboo	2.250030	2.111718	-0.138312
has_secondary_use_use_police	2.238282	2.130435	-0.107847
has_superstructure_timber	2.263274	2.165222	-0.098052
has_secondary_use_agriculture	2.236457	2.264648	0.028191
has_superstructure_mud_mortar_brick	2.235863	2.271212	0.035349
has_superstructure_adobe_mud	2.227718	2.346782	0.119064
has_superstructure_stone_flag	2.230654	2.452554	0.221900
has_superstructure_mud_mortar_stone	1.919407	2.337901	0.418494

Topics/Subjects to talk about:

- Intro
- Earthquake background
- Problem Statement
- Data Background
    - Where the raw data came from
    - Talking about the DataDriven competition, etc.
- Data Feature Description
    - What they mean
    - Mention that categorical variable value types were encrypted as random letters
    - Building IDs also random numbers to keep you from cross-referencing the original raw data
- EDA
    - Data Size & Shape
        - 40 Columns
        - 38 Features
        - 1 Target variable with 3 possible categories
        - Total Building Entries: 347,469
        - Train Entries: 260,601
        - Test Entries: 86,868
    - Already cleaned by competition host
    - Average Damage Grade is **2.238272**
    - Baseline Accuracy: Always guessing Grade 2 has an accuracy of 56.89%
        - Grade 1: 9.64%
        - Grade 2: 56.89%
        - Grade 3: 33.47%
- Visualizations
    - Categorical
        - The struggles of not having category value types for interpreting coefficients.
        - We can still use these variables to classify damage grades, but we don't know which variable is the most impactful.
    - Numerical (Non Binary)
        - Correlations with damage grade (as a number)
    - Numerical (Binary)

- The picture on the right is the average damage grade of the buildings in the dataset when grouped by the two different values of each binary value.
- For example, buildings whose superstructure is made of engineered reinforced concrete had an average damage grade of 1.38, while buildings that didn't have a superstructure made of engineered reinforced concrete had an average damage grade of 2.252
- Talk about which variables are potentially most useful for the model based on the differences in averages (sorted in the difference column)

Jack

## Data and Visualizations

Slide 5: Our dataset comes from a kaggle-esque modeling competition titled "Richter's Predictor: Modeling Earthquake Damage" that's hosted by the website, DrivenData. The data was originally collected by Kathmandu Living Labs and the Central Bureau of Statistics. The training and test data consists of approximately 260,000 and 87,000 unique entries respectively where each entry consists of almost 40 descriptive features about a unique building that was hit by the 2015 Nepal Earthquake.

Slide 6: The variable we're attempting to classify is a grade classification indicating the amount of damage sustained by the building, categorized into grades 1, 2, and 3. 1 indicates low damage, 2 indicates medium damage, and 3 indicates that the building was almost completely destroyed. In our training data, about 56% of the buildings are classified with Grade 2 damage while 33% are classified with Grade 3 levels of destruction.

Slide 7: The building features include a variety of characteristics from the number of families living inside each building and number of floors they originally had, to the materials that the building roof, foundation, and ground floor was made out of. The biggest challenge we encountered with our data analysis is due to the nature of the DrivenData competition. When cleaning the data to provide contestants with training and test data, the authors of the competition intentionally encrypted the value types of every categorical variable as well as the unique building and region identifiers with random letters and numbers, respectively. This was most likely done so that data scientists taking part in the competition would be unable to cross-reference the training data with outside data sources and other features that could facilitate the model training process. We can train the model with these encryptions, but feature interpretation becomes a heck of a lot harder. As a result, while we were able to deduce what subcategories these variables consist of, we were unable to map most subcategories to their encrypted letter except for the ones that have a frequency distribution that makes them stick out like a sore thumb. For example, the feature, foundation type refers to the material with which the building's foundation was constructed. Using raw data from which the DataDriven competition was designed, we know that the possible materials are Mud Mortar with Stone/Brick, Cement with Stone/Brick, Bamboo and Tinder, reinforced concrete, and “other”. When we analyzed the raw data, we noticed that in the complete data set of 700,00+ entries, the percentage of houses with a Mud Mortar foundation is over 80%, while the other four subcategories had frequencies ranging from 1 to 10 percent. Through this, it was safe to assume that our “R” variable could be attributed to Mud Mortar, because the foundation type of 84% of buildings in our training set are classified as “R” with the other categories appearing at a frequency of less than 6%.

Slide 8: Before modeling, we examined each feature to see how average damage grade differed between buildings when grouped by different feature values. In order to compare damage grade between feature sub-categories prior to modeling, I treated the target variable as ordinal as the damage a building sustained is higher as the grade number increases. Our feature types consisted of multi-value categorical variables such as the geographical region a building is located, numerical variables such as the number of families living in a building, and two pre-encoded categorical variables. The two pre-encoded variables consist of the material the building’s superstructure was built with, and whether the building has any secondary uses such as government offices, education, and agriculture. Some variables appeared to have little to no effect on damage grade such as whether a building is used for agriculture or not, while others had noticeable differences such as buildings built from engineered reinforced concrete having an average damage grade that’s about 0.87 less than the buildings built from other materials.

Emily

## Model and Findings
Slide 9: Because our dataset had over 260,000 datapoints, we decided to first work with smaller sample size of 1,000 and 5,000 data points. This allowed us to build and test models succinctly without the lengthy model run times on such a large data set. The models we built on these smaller datasets were Logistic Regression, SVC, Random Forest Classifier, Decision Trees Classifier, GradientBoost Classifier, and KNearestNeighbors Classifier. We chose these as we wanted to test a broad range of classifier model types in order to see which could produce the best F1 score.
As for our metric, we are choosing the F1 score as the best evaluation metric for our model because it is that balanced score between precision and recall, and it helps explain the relationship around all positive predictions and false negatives. It also is beneficial for us because we have moderately imbalanced data. It is more transparent about model performance, for example, if our model was heavily predicting one class over another. By evaluating the F1 score, we can get a more clear picture of model performance on our data.

Slide 10: So, we found that K Nearest Neighbors Classifier and Gradient Boost Classifier fit best on our sample datasets. Although these models fit well on the training set, we noted overfitting on the sample test sets. However, we decided to move forward with using these models on our big dataset as they showed promise for a larger scale model.

Our KNN model fit the big dataset well with a 0.71 test F1 score. The GBC fit the big dataset slightly worse with a 0.68 for both train and test F1 scores. Between these and because we chose F1 to be our evaluation metric, our final model choice is KNN. This intuitively makes sense as we can see structures built in clusters like cities and neighborhoods; it makes sense that a model looking around a certain data point would find similarities. While the GradientBoost performed well on the data, the KNN just slightly outperformed and that was the ultimate decider on our final model. While we were mostly focused on the F1 score, I will also note that the accuracy score for our model was 72%.

Slide 11: Our model also outperformed the baseline model of 56% predicting all medium level damage. While the model still heavily predicted medium level damage, it predicted more accurately at 72% on the test data. One key part of building our model was tuning hyperparameters. We found that using a neighbor count of 8 was the best hyperparameter. Since our dataset was so large at over 260,000 data points, having a larger neighbor count is beneficial to working through the entirety of the data. Finally, even though KNN over fit on the smaller datasets, it did not show major overfitting on the larger dataset. We attributed this to the larger number of datapoints to train on as compared to the smaller dataset.

Marva