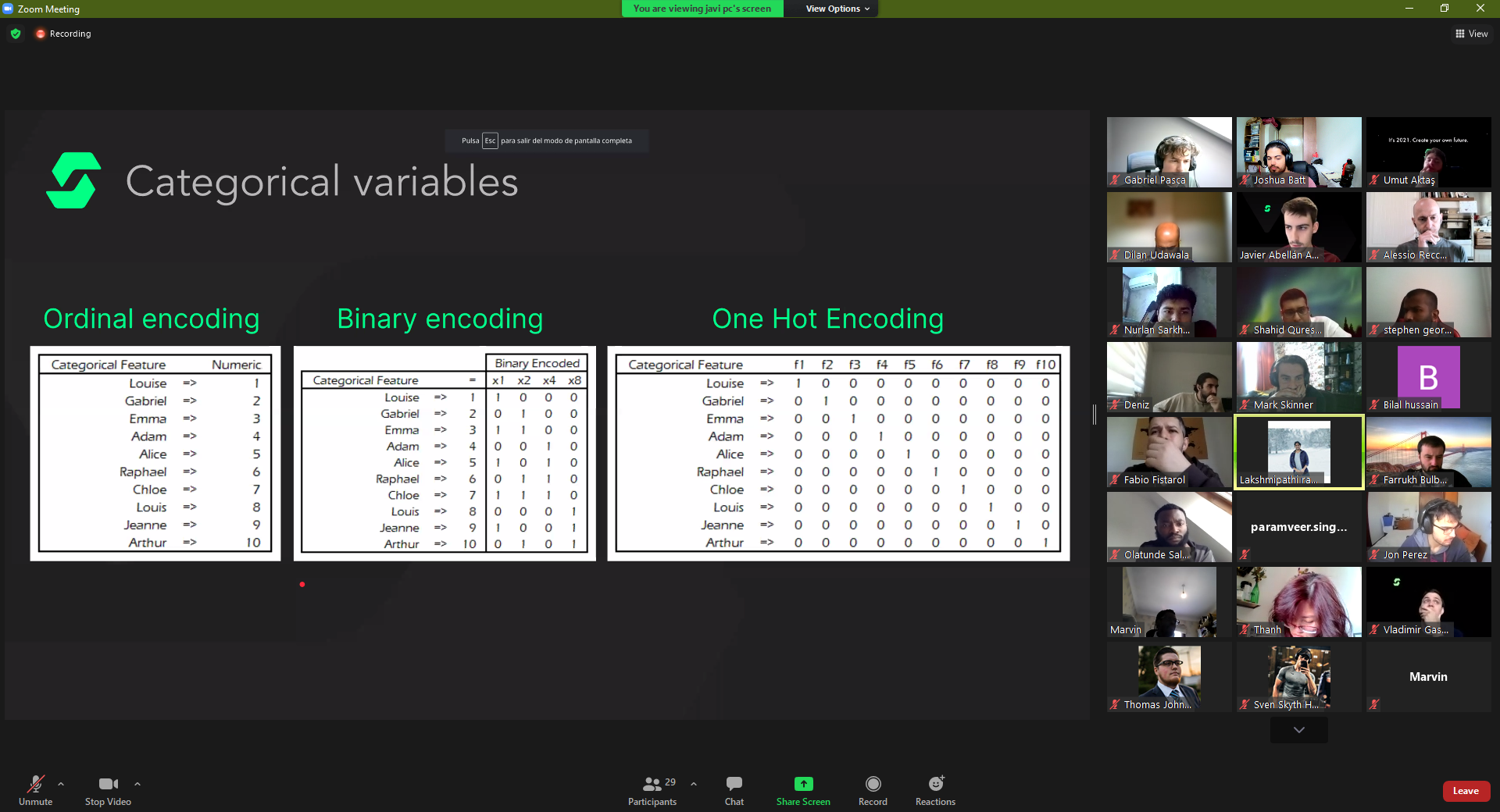
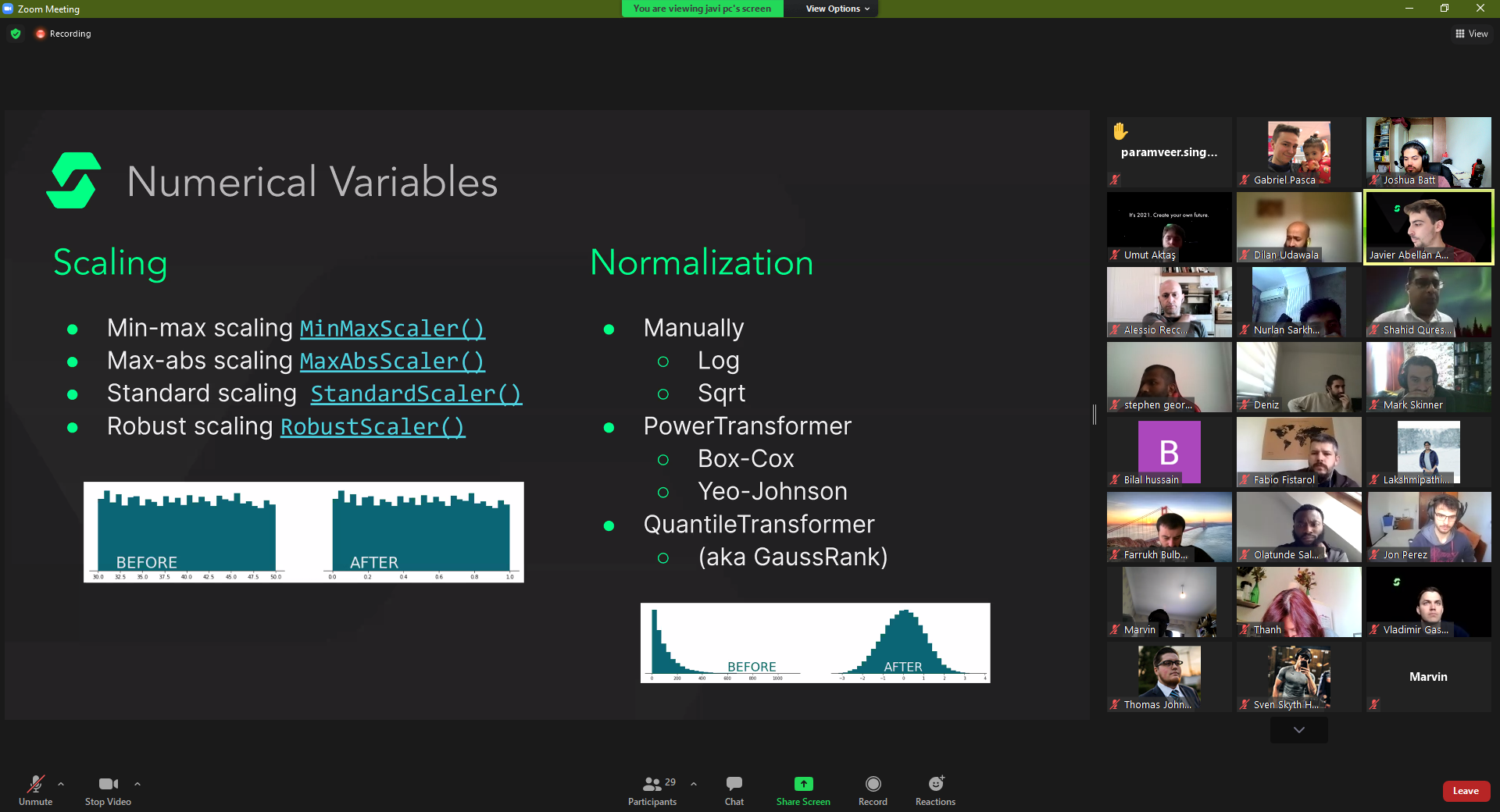
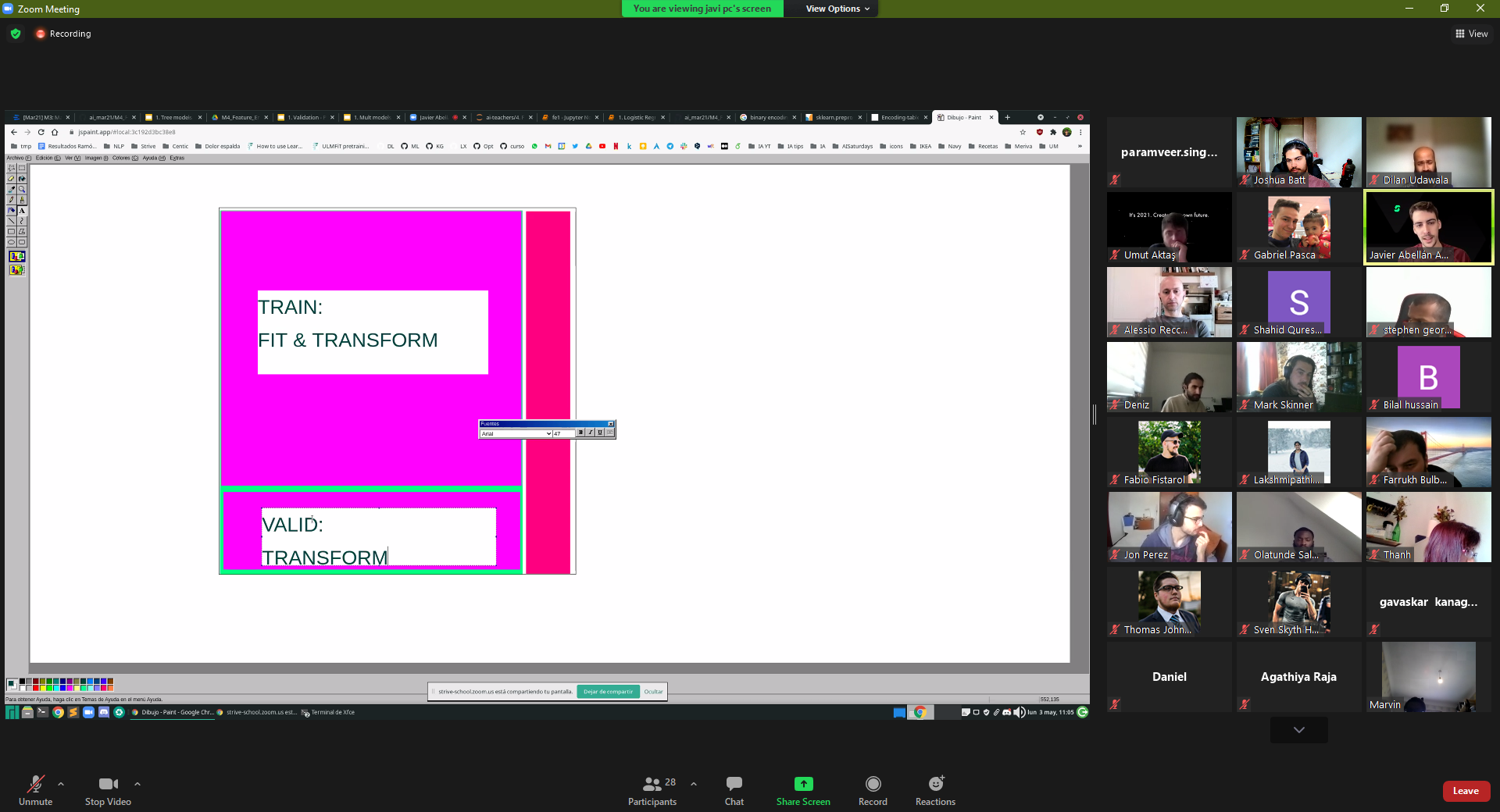
**M4 D1**

**Trees (decision trees, random forest, and more…):**

* Gradient Boosting (GBM)
  + The same as Random Forest but, improves on the error of the previous trees.
  + Limit the depth of each tree to cap computational power.
  + **Most powerful type of model**.
  + Random forest trees are independent, but gradient boosting is improving on the previous trees.
  + Boosting means improving sequentially
  + Hyperparameter are important to not overfit.
    - Add values in a logarithmic scale: add 0.1, if its too much then add 0.01. etc…
* Categorical variables
  + 
  + Ordinal encoding? – WHAT IS IT?
  + Binary encoding
    - Label the quantities that things can come up in columns.
    - Putting the 1s in place of quantities will result in binary looking numbers.
  + One hot encoding
    - Giving each entry its own column.
    - Takes too many computations to give everything its own column.
* Scaling vs Normalizing:
  + 
    - Scaling does not change the data’s plot.
    - Normalization changes the data’s plot.

**Workflow Tips:**

* df.isnull().sum
  + Shows the null values in each column.
  + df.isnull().sum / len(df) = Normalize data
* train\_test\_split
  + test\_size = 0.number
    - Whole number = rows
    - Decimal = percentage
  + Stratify = y – GOOGLE WHY?
  + Random state = Any number you want, just keep using it
* .fit & .transform for training data
  + Only .transform for test data
  + 
    - Pink is x data (The columns we will use to make our predictions)
    - Right side red/pink is y data (The data we are trying to predict)