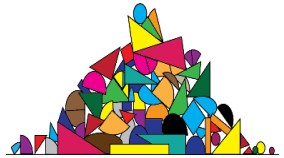
Train In Data

Putting it all together

Typical machine learning process

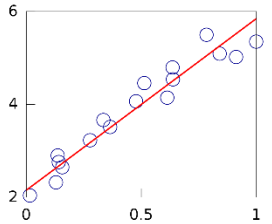
Data

Data Analysis



Feature Engineering

Variable selection

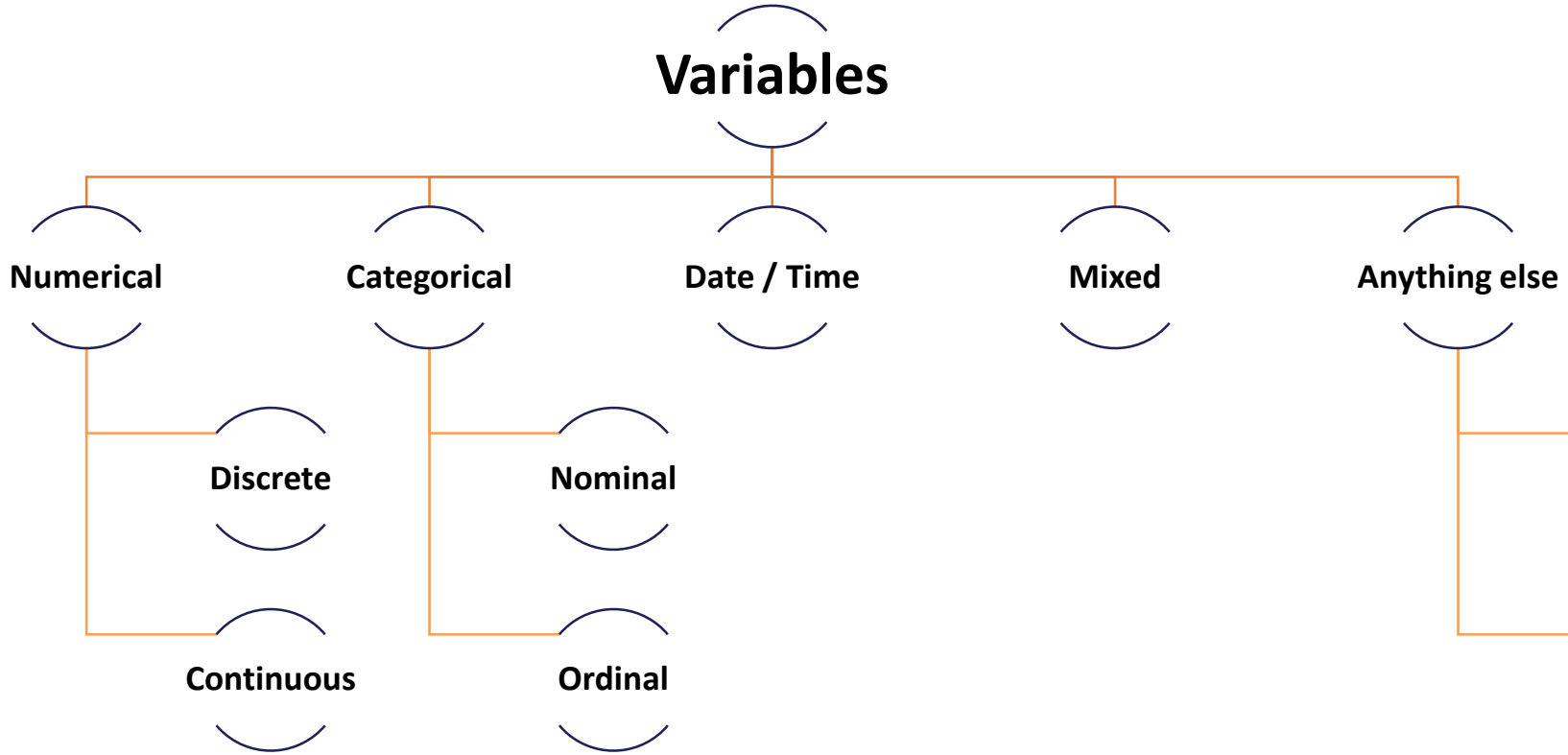


Machine Learning Model building

Deployment

Data Analysis ➔ Variable Types

Data Analysis



Data Analysis ➔ Variable Characteristics

Data Analysis

Variable Characteristics

01 **MISSING DATA**
Lack of information for some observations within a variable.

02 **CATEGORICAL VARIABLES**
Variables contain strings instead of numbers

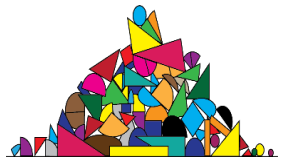
03 **LINEAR MODEL ASSUMPTIONS**
Do the variables fulfill those assumptions?

04 **DISTRIBUTIONS**
Normal, skewed and others.

05 **OUTLIERS**
Unusual or unexpected values

06 **FEATURE MAGNITUDE**
Scale of the different features

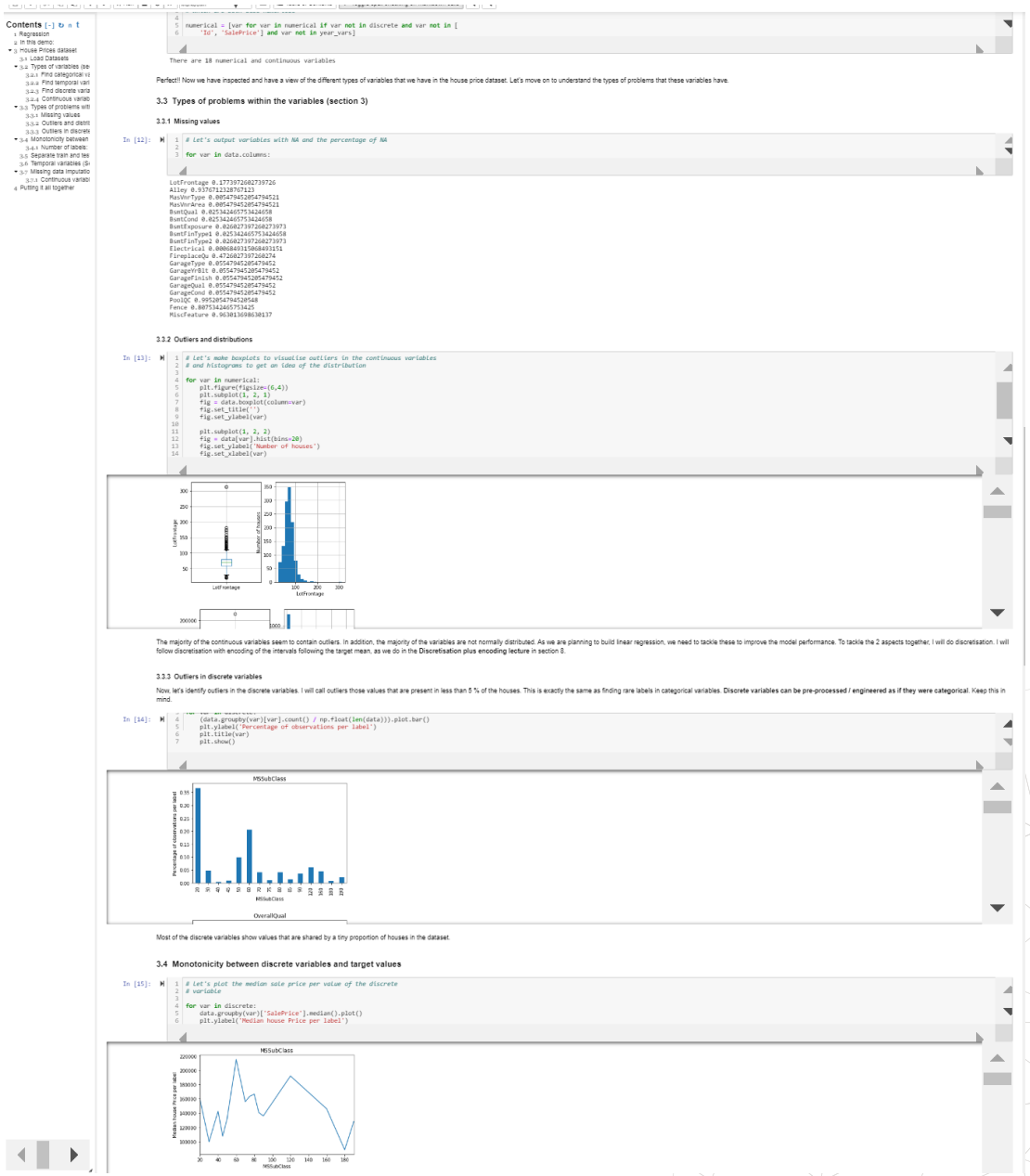
Feature engineering steps



Feature Engineering

- Feature Creation:
  - Extracting Features from Dates
  - Extracting Features from Mixed Variables
    - Missing Data Imputation
      - Categorical Variable Encoding
        - Numerical Variable Transformation
          - Discretisation
            - Outlier Handling
              - Feature Scaling
                - And more…

Jupyter notebook



- Hard to understand
- Hard to reproduce
- Hard to score new data
- Hard to deploy

Pipeline

- Clear
- Concise
- Reproducible
- Able to score new data
- Easy to deploy

Pipeline

```
price_pipe = pipe([
    # add a binary variable to indicate missing information for the 2 variables below
    ('continuous_var_imputer', msi.AddNaNBinaryImputer(variables = ['LotFrontage', 'GarageYrBlt'])),

    # replace NA by the median in the 3 variables below, they are numerical
    ('continuous_var_median_imputer', msi.MeanMedianImputer(imputation_method='median', variables = ['LotF
rontage', 'GarageYrBlt', 'MasVnrArea'])),

    # replace NA by adding the label "Missing" in categorical variables (transformer will skip those varia
bles where there is no NA)
    ('categorical_imputer', msi.CategoricalVariableImputer(variables = categorical)),

    # there were a few variables in the submission dataset that showed NA, but these variables did not sho
w NA in the train set.
    # to handle those, I will add an additional step here
    ('additional_median_imputer', msi.MeanMedianImputer(imputation_method='median', variables = numerical
)),

    # disretise numerical variables using trees
    ('numerical_tree_discretiser', dsc.DecisionTreeDiscretiser(cv = 3, scoring='neg_mean_squared_error', v
ariables = numerical, regression=True)),

    # remove rare labels in categorical and discrete variables
    ('rare_label_encoder', ce.RareLabelCategoricalEncoder(tol = 0.03, n_categories=1, variables = categori
cal+discrete)),

    # encode categorical variables using the target mean
    ('categorical_encoder', ce.MeanCategoricalEncoder(variables = categorical+discrete))
     ])
```

Pipeline

```
# train the pipeline
price_pipe.fit(X_train, y_train)

# score data
price_pipe.transform(X_train)
price_pipe.transform(X_test)

price_pipe.transform(live_data)
```

Train In Data

THANK YOU

www.trainindata.com