Feature Creation and Feature Engineering is one of the most important tasks in machine learning since it hugely impacts model performance. This also holds for deep learning, although to a lesser extent. Features can be changed or new features can be created from existing ones

The following none exhaustive list gives you some guidelines for feature transformation:

- Imputing
  Some algorithms are very sensitive to missing values. Therefore, imputing allows for filling of empty fields based on its value distribution

- Imputed time-series quantization
  Time series often contain streams with measurements at different timestamps. Therefore, it is beneficial to quantize measurements to a common "heart beat" and impute the corresponding values. This can be done by sampling from the source time series distributions on the respective quantized time steps

- Scaling / Normalizing / Centering
  Some algorithms are very sensitive differences in value ranges for individual fields. Therefore, it is best practice to center data around zero and scale values to a standard deviation of one

- Filtering
  Sometimes imputing values doesn't perform well, therefore deletion of low quality records is a better strategy

- Discretizing
  Continuous fields might confuse the model, e.g. a discrete set of age ranges sometimes performs better than continuous values, especially on smaller amounts of data and with simpler models

The following none exhaustive list gives you some guidelines for feature creation:

- One-hot-encoding
  Categorical integer features should be transformed into "one-hot" vectors. In relational terms this results in addition of additional columns – one columns for each distinct category

- Time-to-Frequency transformation
  Time-series (and sometimes also sequence data) is recorded in the time domain but can easily transformed into the frequency domain e.g. using FFT (Fast Fourier Transformation)

- Month-From-Date
  Creating an additional feature containing the month independent from data captures seasonal aspects. Sometimes further discretization in to quarters helps as well

- Aggregate-on-Target
  Simply aggregating fields the target variable (or even other fields) can improve performance, e.g. count number of data points per ZIP code or take the median of all values by geographical region

As feature engineering is an art on itself, this list cannot be exhaustive. It's not expected to become an expert in this topic at this point. Most of it you'll learn by practicing data science on real projects and talk to peers which might share their secrets and tricks with you.

Please transform your data set accordingly and add all code to the Feature Creation asset deliverable. Please comply with the naming convention documented in the process model.