Chapter 3: Feature Engineering

3.1. Introduction to Feature Engineering

Feature engineering refers to the process of transforming raw data into relevant and meaningful input features standardised to be used by the machine learning models to learn useful patterns and then make accurate predictions. This involves various methods, such as: null value treatment, garbage value treatment, sanity checks, encoding, scaling, selection of the most relevant data, discarding insignificant data etc.

The NPCYF Feature Engineering module handles these tasks in six steps, mentioned chronologically as follows:

1. Feature Master
2. Training Feature Set
3. Training Set Data
4. Training Dataset Collection
5. Training Dataset Analysis
6. Training Dataset Operations

3.2. Purpose and Scopes

3.2.1 Purpose of Feature Engineering

- **Model Accuracy and Performance**
- **Data Quality**
- **Interpretability**
- **Overfitting Reduction**
- **Model Training Efficiency**

3.2.2. Scope of Feature Engineering

- **Feature Creation**
- **Feature Transformation**
- **Feature Selection**
- **Missing Value Treatment**
- **Encoding**
- **Scaling and Normalization**

3.3. NPCYF Feature Engineering Steps

3.3.1. Feature Master

This section is, for all intents and purposes, perhaps the most important and valuable section of NPCYF's Feature Engineering module. Because, this section contains the comprehensive masterlist of all possible features that could be deemed meaningful and/or necessary to build any kind of model for the purpose of any kind of prediction and/or forecast using the NPCYF platform.

This is necessary, because it is quite useful to have a well-structured, well thought out, and carefully curated masterlist of features always accessible at hand. On one hand, this rids the end users without admin privilege of the hassle and headaches of having to start from scratch and create their own feature masterlist - they can simply select any subset of features already available in the Feature Master to create their own, model-specific, feature set for training and prediction.

On the other hand, for the end users *with* admin privilege, it is also beneficial to have access to a pre-existing masterlist of features, so as to gain an overall idea about what kind of features, along with what kind of temporal, spatial intervals and aggregation methods could prove to be useful if added to the Feature Master.

To know how to insert a new feature into the Feature Master, please refer to the *tutorial*.

3.3.2. Training Feature Set

Not all features are equally important for all models and/or all predictions. Nor are they equally useful. Thus, it is necessary to create specific feature sets to train different models for different prediction purposes.

The Training Feature Set section of NPCYF's Feature Engineering module has been designed to help the user not only create model-specific training feature sets but also edit, reset and/or delete them if necessary. This section also makes it easy to keep track of all the different feature sets created to train different models for different prediction purposes.

The user can do so by first defining and describing the feature set they want to create - for the sake of convenience - and then adding as many features from the features available in the Feature Master as wanted/needed to that feature set.

🔎 ***Technical Insight***

The most common techniques used to select useful features for a model are - Filter Methods, Wrapper methods, Embedded Methods and Dimensionality Reduction methods. For Further Details, refer to Appendix II

3.3.3. Training Set Data

Simply specifying a training feature set to obtain desired predictions and/or forecasts is not sufficient. In order to train models, actual data must be generated that strictly adheres to the structure and semantics defined by the selected feature set.

The generation of such a dataset involves dynamically constructing the appropriate SQL query based on the chosen feature set and dataset-specific filters (such as crops, seasons, locations, and time range), followed by executing that query. Upon execution, the resulting dataset is materialized as a SQL view for further querying and can optionally be exported or stored in external storage formats such as CSV or Iceberg tables.

The *Training Set Data* section of NPCYF's Feature Engineering module fulfills this role. Each time a training feature set is selected and dataset-specific parameters are provided, this section generates exactly one concrete dataset derived from that specific feature set under the specified filters.

> 🔎 ***Technical Insight***
>
> *The "Generate" button, when clicked, takes into account the filters chosen (e.g. crop, year), and then generates the sql query which will be executed to create the dataset abiding by the feature and filter specifications of the selected feature set.*
>
> *The "materialize" button, when clicked, then executes that generated sql query to create a concrete dataset and then store it in the form of an SQL view and MinIO URL so that the user can later on see the result if desired and/or export it in the form of csv file(s).*
>
> *For further details, see Appendix II*

3.3.4. Training Dataset Collection

While a *Training Set Data* represents a single concrete dataset generated from a specific training feature set and a fixed set of filters, real-world model development and experimentation often require working with multiple related datasets together. These datasets may represent different time ranges, geographic regions, crops, seasons, or experimental splits (such as training, validation, and testing datasets).

The *Training Dataset Collection* section of NPCYF’s Feature Engineering module serves as an organizational and management layer over individual training datasets. It allows multiple *Training Set Data* entities to be grouped together under a single logical collection, enabling them to be treated as a coherent unit for downstream model training, evaluation, and experimentation.

> 🔎 ***Technical Insight***
>
> *A training dataset collection does not generate new data on its own. Instead, it references and aggregates already-generated training datasets, preserving their individual definitions while providing a higher-level structure for experiment management, dataset versioning, and reproducibility. Each collection can contain one or more training datasets, with enforced uniqueness to prevent accidental duplication within the same collection.*

3.3.5. Training Dataset Analysis

A crucial step of data science is to use statistical and visual tools to analyse the available data to understand its main features, discover patterns, notice anomalies, test hypotheses and check assumptions. This is necessary to gain insight about how a data might behave or what kind of modifications might be necessary before it can be fed to a model for training purposes.

The “Training Dataset Analysis” section of NPCYF’s Feature Engineering module serves that very purpose. In this section, it’s been made possible to filter a dataset by the training feature set it’s under and the training dataset collection it’s part of, and then fetch its analysis metrics.

It’s also possible to visualize the analysis metrics in terms of various graphical representations, and hence get a further idea of the relationships between the features within the datasets.

> 🔎 ***Technical Insight***
>
> *The process of analysing a dataset before feeding it to train a model is called Exploratory Data Analysis. It involves summarizing data, visualizing relationships, identifying errors, and generating new questions, ultimately guiding further analysis and decision-making.*

3.3.5. Training Dataset Operations

It is sometimes necessary to modify the feature columns of a dataset to make it suitable for model training and/or prediction/forecasting.

Various operations can be performed on different columns based on their nature, requirements, and relationship with each other. The “Training Dataset Operation” section of NPCYF’s Feature Engineering module helps perform those operations.

This way, it’s made certain on the platform that the datasets are suitable for moving forward with and to train models to get accurate predictions/forecasts.

3.4. Tutorial - Feature Engineering

3.4.1. Tutorial - Part 1: Training Feature Set

Click on “Training Feature Set” on the dashboard, which will bring you to the “Training Feature Set” page.

3.4.1.1. Step 1 - Definition

Define your training feature set by naming and describing it. Fill in the “Name” and “Description” form, check the “is Active” checkbox, and then click on the Add button to add your new training set definition.

.

Upon success, a pop-up will appear to let the user know that the new definition for the training feature set has been added to the table.

You have now successfully defined your training feature set.

3.4.1.2. Step 2 - Selecting Features for the Training Feature Set

Upon clicking the “Feature” option, we arrive at the corresponding section where we must select features for our training set.

From the “Select Training Set” drop-down list, select your newly defined training set.

From the “Select Category” drop-down list, choose the feature category you want.

Upon selecting your desired feature category, a third drop-down list will appear. The “Select Feature” drop-down displays all features available in the feature master under the specified category. Select the desired feature from the list, check the “is Target” checkbox if this is your target feature (i.e., the feature you want a prediction for).

After selecting the feature, click on the “Add” button, and the feature will be added to your training feature set and will be visible in the table. A pop-up will also appear upon successful addition.

Add as many features as you want for your training feature set.

3.4.1.3. Step 3 - Temporal Interval for Each Chosen Feature

Upon clicking on “Temporal Interval”, you’ll reach the corresponding section, where you need to specify the time interval of each of the features in your training feature set.

From the “Select Training Set” drop-down list, choose your training feature set.

Upon selecting your training set, a second drop-down list will appear. This “Select Feature” list contains all the features you’ve chosen from the feature master for your training feature set. Select a feature from this list.

After choosing a feature, three more drop-down lists will appear for you to specify the start and end of the interval range, and the base year.

Specify the start and end of your desired interval range, and the base year. Here, the base year signifies that, for a dataset containing data of a feature from one year to another, whether you want to prioritize the beginning or the end of that interval.

After this, click on the “Add” button to submit the temporal interval for your feature. The interval will be added to the table, along with a pop-up showing the success of the procedure.

Do the same for each feature in your training feature set.

Now, you have completed configuring your training feature set and are ready to move on to the next part.

3.4.2. Tutorial - Part 2: Training Set Data

From the dashboard, if you click on “Training Set Data”, it takes you to the corresponding page. We will treat this page, for all intents and purposes, as if it’s divided into two partitions. This is purely for the convenience and ease of understanding how this page works.

The first Partition contains a drop-down list, “Select Training Set”, a table, and three buttons below it - “Delete”, “Generate”, and “Materialise”.

The second partition has multiple drop-down lists and selectors, and two buttons at the very bottom: “Add” and “Reset”.

3.4.2.1. Step 1 - Training Set Selection

From the “Select Training Set” drop-down list, choose a training feature set that has been defined and added to the platform in the previous part, “Training Feature Set”.

Upon selecting a training feature set from the list, all the datasets generated for that training set as of yet will appear in the table.

In the next step, we will show how to generate a new dataset catered to the chosen training feature set and add to the table.

3.4.2.2. Step 2 - Data Selection

In the bottom partition, fill up all the forms from top to bottom, and left to right, with all the desired values/ranges/choices suitable for your model and forecasting.

Finally, click the “Add” button to add the specified dataset to the table. Upon successful addition, a pop-up message will appear informing of the completion of the process.

3.4.2.3. Step 3 - Dataset Generation, Materialisation, and Viewing

Now that the dataset specification has been added, come back to the table and select the newly added training set data.

Now, this training set data needs to be generated by putting together all the feature columns from various scraped datasets available on the platform. Click on the "Generate" button, and a pop-up will appear informing you that the generation has started in the background.

After the dataset has been generated, click the “Materialise” button. This will make the dataset ready for viewing. A similar pop-up will appear, informing you that the process has started in the background.

After the dataset has been successfully generated and materialised, both the boxes will be checked in the table, and a view button will appear, upon clicking which the dataset will be visible along with an option to download it as well.

Now, you’ve successfully generated your own training dataset catered to your feature set. We can now move on to the next part of this walkthrough.

3.4.3. Tutorial - Part 3: Training Dataset Collection

On the dashboard, click on "Training Dataset Collection" to reach the corresponding page

.

This page is divided into two sections: "Definition" and "Dataset Items". In the "Definition" section, the dataset collection has to be named and described. And then in the "Dataset Items" section, we gather all the training datasets we want under that collection.

The purpose of having a collection of training datasets is that we can have the same training feature set for different target variables. And for each target variable, we need to generate a new training dataset catering to the training set feature specifications. Having all of them under one single collection makes accessing them easy and convenient.

3.4.3.1. Step 1 - Definition

Fill up the form by adding a suitable name and description for your dataset collection.

After this, click the “Add” button to include your dataset collection in the table. Upon successful addition, a pop-up will appear with a success message.

3.4.3.2. Step 2 - Dataset Items

Click on “Dataset Items”, and it will take you to the relevant section.

From the “Select Collection” drop-down list, pick the collection you just defined.

From the “Select Training Set” drop-down list, pick your relevant training set. Upon selecting one, a selector list will appear right below, containing all the datasets catering to your specified training feature set. Choose as many datasets as you want to add to your collection.

Click on the “Add” button after picking your datasets, and they will be added to your collection. A pop-up will appear informing you of the successful addition.

You have now successfully created your dataset collection and can proceed with the next part.

3.4.4. Tutorial - Part 4: Training Dataset Analysis

On the dashboard, click on “Training Dataset Analysis” to reach the corresponding page.

Here, we will analyse the datasets under a specific training feature set, fetch the analysis metrics, and then visualize the graphical plottings of the features and their relationships with each other.

3.4.4.1. Step 1 - Selection of the Dataset

From the “Select Training Set” drop-down list, choose your desired training feature set. Then, from the “Select Dataset” drop-down list, choose a dataset collection under that training feature set. Upon your selection, all the datasets in that collection will appear in the table below.

Now, click on the “Analyse” button to start the analysis. A pop-up will appear when the analysis metrics have been fetched successfully. Now we are ready to visualize the analysis of our dataset collection.

3.4.4.2. Step 2 - Visualisation

Click on “Visualisation”, and select your training feature set and corresponding dataset once again. It will appear in the table below.

Now select a row from this dataset and click the “Visualize” button.

Now, a dialogue box will appear with various visualisation options. Select your preferred option and visualize your dataset in a graphical representation.

Now, the analysis of the datasets is complete, and we are ready to proceed with the final part of feature engineering: applying transformation operations on the datasets.

3.4.5. Tutorial - Part 5: Training Dataset Operations

From the dashboard, click on the “Training Dataset Operations” card to reach the corresponding page.

3.4.5.1. Step 1 - Selection of the Dataset Collection

From the dropdown list, select a dataset collection.

Upon selection, all the individual datasets available under that collection will be available on the screen.

3.4.5.2. Step 2 - Selection of Dataset

From the list of datasets, select a dataset and click on the tab.

3.4.5.3. Step 3 - Selection of Transformation Operation

From the “Select a transformation method” drop down list, pick the transformation operation that you want to perform on the columns of your chosen dataset.

Finally, click on the “Preview Transformation” button at the bottom right of the screen to show how your dataset would look like after applying the chosen transformation operation.

If you’re satisfied with the results, then click on the “Confirm Transformation” button that appears at the bottom-right of the screen after successful preview generation.

This will actually apply your chosen transformation operation on the selected dataset columns and alter them accordingly.