What is Feature Engineering — Importance, Tools and Techniques for Machine Learning

Feature engineering techniques for machine learning are a fundamental topic in machine learning, yet one that is often overlooked or deceptively simple.

Harshil Patel · Follow
Published in Towards Data Science
11 min read · Aug 30, 2021

Listen Share ••• More

Open in app ↗

Search

A

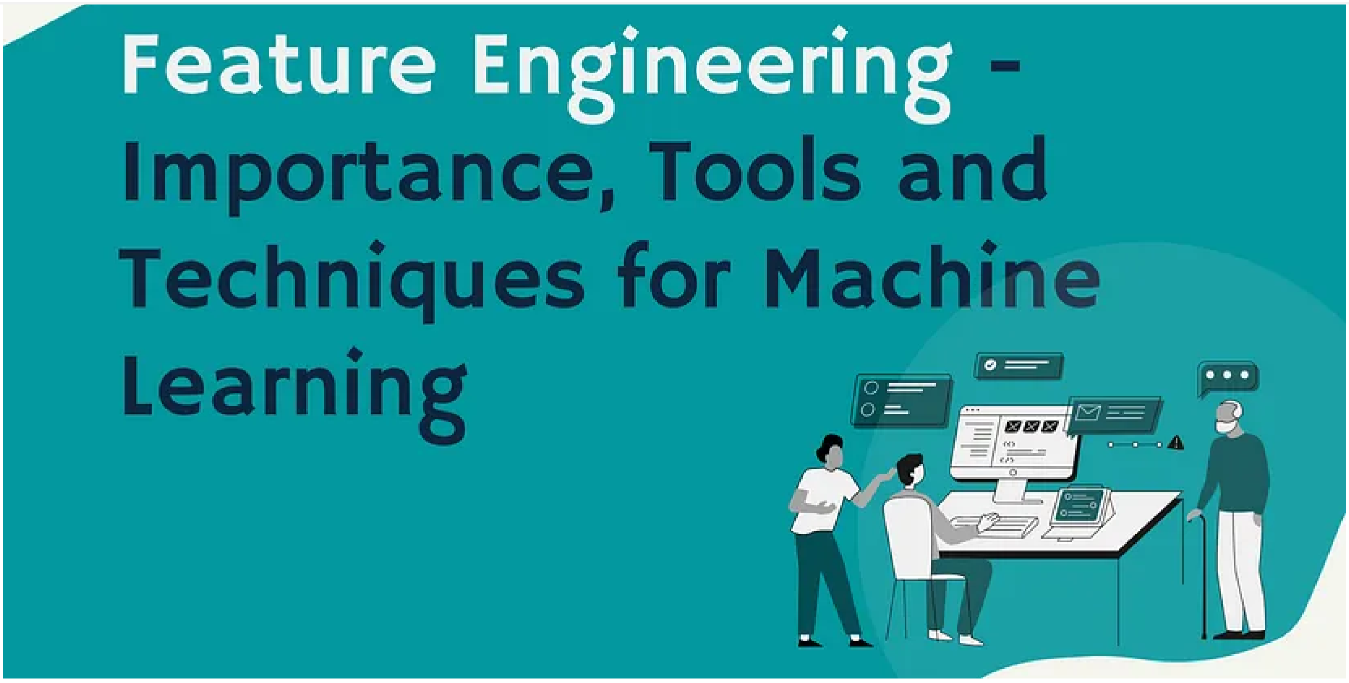


Image By Author

Feature engineering is the process of selecting, manipulating, and transforming raw data into features that can be used in supervised learning. In order to make machine learning work well on new tasks, it might be necessary to design and train better features. As you may know, a "feature" is any measurable input that can be used in a predictive model — it could be the color of an object or the sound of someone's voice. Feature engineering, **in simple terms, is the act of converting raw observations into desired features using statistical or machine learning approaches.**

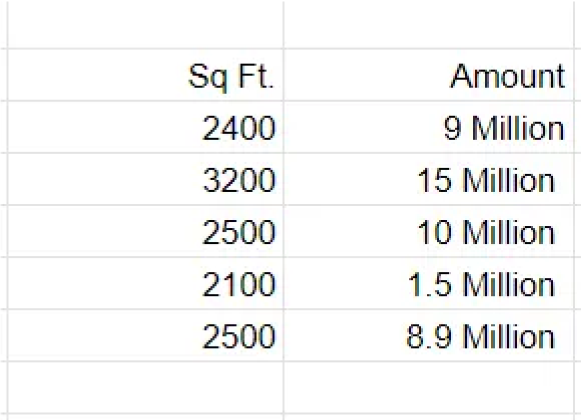
In this article we will see :

- What is Feature engineering,
- Importance of Feature Engineering,
- Feature Engineering Techniques for Machine Learning,
- Few Best tools for feature engineering.

What is Feature Engineering

Feature engineering is a machine learning technique that leverages data to create new variables that aren't in the training set. It can produce new features for both supervised and unsupervised learning, with the goal of **simplifying and speeding up data transformations** while also **enhancing model accuracy**. Feature engineering is required when working with machine learning models. Regardless of the data or architecture, a terrible feature will have a direct impact on your model.

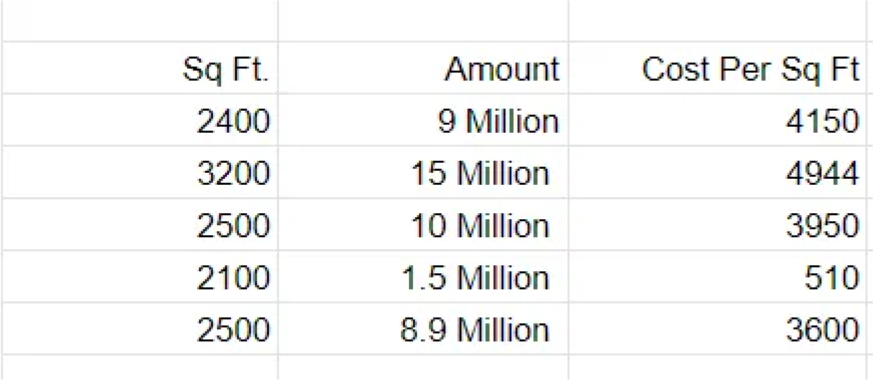
Now to understand it in a much easier way, let's take a **simple example**. Below are the prices of properties in x city. It shows the area of the house and total price.



Sq Ft.	Amount
2400	9 Million
3200	15 Million
2500	10 Million
2100	1.5 Million
2500	8.9 Million

Sample Data

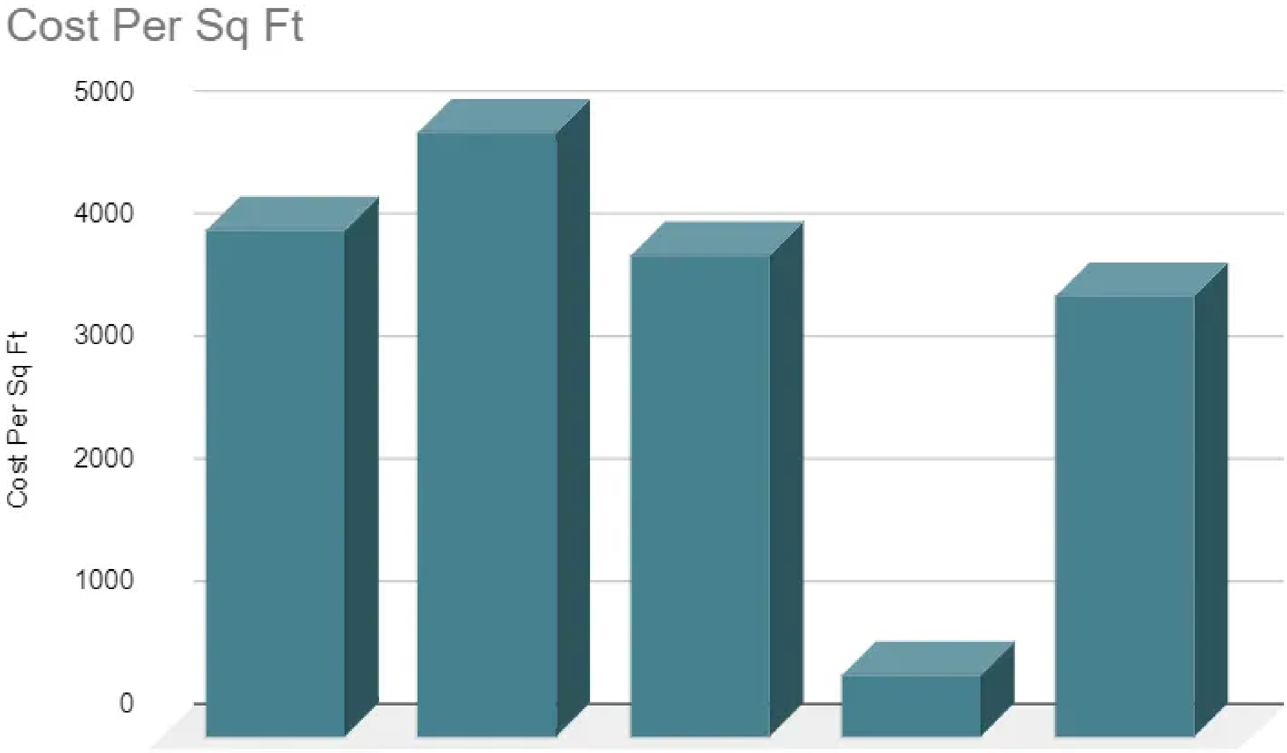
Now this data might have some errors or might be incorrect, not all sources on the internet are correct. To begin, we'll add a new column to display the cost per square foot.



Sq Ft.	Amount	Cost Per Sq Ft
2400	9 Million	4150
3200	15 Million	4944
2500	10 Million	3950
2100	1.5 Million	510
2500	8.9 Million	3600

Sample Data

This new feature will help us understand a lot about our data. So, we have a new column which shows cost per square ft. There are **three main ways** you can find any error. You can use **Domain Knowledge** to contact a property advisor or real estate agent and show him the per square foot rate. If your counsel states that pricing per square foot cannot be less than 3400, you may have a problem. The data can be **visualised**.



When you plot the data, you'll notice that one price is significantly different from the rest. In the **visualisation method**, you can readily notice the problem. The third way is to use **Statistics** to analyze your data and find any problem. Feature engineering consists of various process -

- **Feature Creation**: Creating features involves creating new variables which will be most helpful for our model. This can be adding or removing some features. As we saw above, the cost per sq. ft column was a feature creation.

- **Transformations**: Feature transformation is simply a function that transforms features from one representation to another. The goal here is to plot and visualise data, if something is not adding up with the new features we can reduce the number of features used, speed up training, or increase the accuracy of a certain model.

- **Feature Extraction**: Feature extraction is the process of extracting features from a data set to identify useful information. Without distorting the original relationships or significant information, this compresses the amount of data into manageable quantities for algorithms to process.

- **Exploratory Data Analysis :** Exploratory data analysis (EDA) is a powerful and simple tool that can be used to improve your understanding of your data, by exploring its properties. The technique is often applied when the goal is to create new hypotheses or find patterns in the data. It's often used on large amounts of qualitative or quantitative data that haven't been analyzed before.

- **Benchmark** : A Benchmark Model is the most user-friendly, dependable, transparent, and interpretable model against which you can measure your own. It's a good idea to run test datasets to see if your new machine learning model outperforms a recognised benchmark. These benchmarks are often used as measures for comparing the performance between different machine learning models like neural networks and support vector machines, linear and non-linear classifiers, or different approaches like bagging and boosting. To learn more about feature engineering steps and process, check the links provided at the end of this article. Now, let's have a look at why we need feature engineering in machine learning.

Importance Of Feature Engineering

Feature Engineering is a very important step in machine learning. Feature engineering refers to the process of designing artificial features into an algorithm. These artificial features are then used by that algorithm in order to improve its performance, or in other words reap better results. Data scientists spend most of their time with data, and it becomes important to make models accurate.

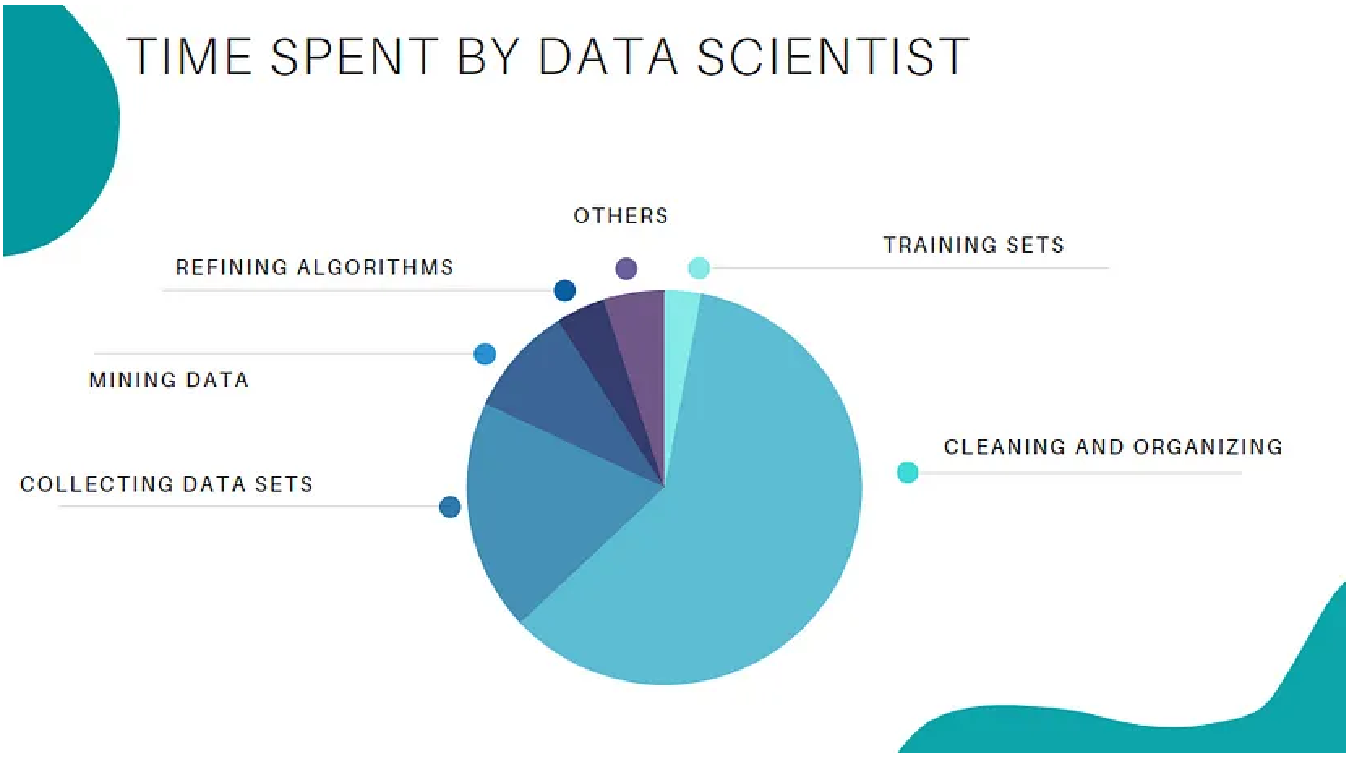


Image By Author

When feature engineering activities are done correctly, the resulting dataset is optimal and contains all of the important factors that affect the business problem. As a result of these datasets, the most accurate predictive models and the most useful insights are produced.

Feature Engineering Techniques for Machine Learning

Lets see a few feature engineering best techniques that you can use. Some of the techniques listed may work better with certain algorithms or datasets, while others may be useful in all situations.

1.Imputation

When it comes to preparing your data for machine learning, missing values are one of the most typical issues. Human errors, data flow interruptions, privacy concerns, and other factors could all contribute to missing values. Missing values have an impact on the performance of machine learning models for whatever cause. The main goal of imputation is to handle these missing values. There are two types of imputation :

- **Numerical Imputation**: To figure out what numbers should be assigned to people currently in the population, we usually use data from completed surveys or censuses. These data sets can include information about how many people eat different types of food, whether they live in a city or country with a cold climate, and how much they earn every year. That is why numerical imputation is used to fill gaps in surveys or censuses when certain pieces of information are missing.

```
#Filling all missing values with 0

data = data.fillna(0)
```

- **Categorical Imputation:** When dealing with categorical columns, replacing missing values with the highest value in the column is a smart solution. However, if you believe the values in the column are evenly distributed and there is no dominating value, imputing a category like “Other” would be a better choice, as your imputation is more likely to converge to a random selection in this scenario.

```
#Max fill function for categorical columns

data[‘column_name’].fillna(data[‘column_name’].value_counts().idxmax(),
inplace=True)
```

2.Handling Outliers

Outlier handling is a technique for removing outliers from a dataset. This method can be used on a variety of scales to produce a more accurate data representation. This has an impact on the model’s performance. Depending on the model, the effect could be large or minimal; for example, linear regression is particularly susceptible to outliers. This procedure should be completed prior to model training. The various methods of handling outliers include:

1. **Removal**: Outlier-containing entries are deleted from the distribution. However, if there are outliers across numerous variables, this strategy may result in a big chunk of the datasheet being missed.

2. **Replacing values**: Alternatively, the outliers could be handled as missing values and replaced with suitable imputation.

3. **Capping**: Using an arbitrary value or a value from a variable distribution to replace the maximum and minimum values.

4. **Discretization :** Discretization is the process of converting continuous variables, models, and functions into discrete ones. This is accomplished by constructing a series of continuous intervals (or bins) that span the range of our desired variable/model/function.

3.Log Transform

Log Transform is the most used technique among data scientists. It's mostly used to turn a skewed distribution into a normal or less-skewed distribution. We take the log of the values in a column and utilise those values as the column in this transform. It is used to handle confusing data, and the data becomes more approximative to normal applications.

```
//Log Example

df[log_price] = np.log(df['Price'])
```

4.One-hot encoding

A one-hot encoding is a type of encoding in which an element of a finite set is represented by the index in that set, where only one element has its index set to "1" and all other elements are assigned indices within the range [0, n-1]. In contrast to binary encoding schemes, where each bit can represent 2 values (i.e. 0 and 1), this scheme assigns a unique value for each possible case.

5.Scaling

Feature scaling is one of the most pervasive and difficult problems in machine learning, yet it's one of the most important things to get right. In order to train a predictive model, we need data with a known set of features that needs to be scaled up or down as appropriate. This blog post will explain how feature scaling works and why it's important as well as some tips for getting started with feature scaling.

After a scaling operation, the continuous features become similar in terms of range. Although this step isn't required for many algorithms, it's still a good idea to do so. Distance-based algorithms like k-NN and k-Means, on the other hand, require scaled continuous features as model input. There are two common ways for scaling :

Normalization : All values are scaled in a specified range between 0 and 1 via normalisation (or min-max normalisation). This modification has no influence on the feature's distribution, however it does exacerbate the effects of outliers due to lower standard deviations. As a result, it is advised that outliers be dealt with prior to normalisation.

Standardization: Standardization (also known as z-score normalisation) is the process of scaling values while accounting for standard deviation. If the standard deviation of features differs, the range of those features will likewise differ. The effect of outliers in the characteristics is reduced as a result. To arrive at a distribution with a 0 mean and 1 variance, all the data points are subtracted by their mean and the result divided by the distribution's variance.

Learn More about Feature Engineering Techniques

Few Best Feature Engineering Tools

There are many tools which will help you in automating the entire feature engineering process and producing a large pool of features in a short period of time for both classification and regression tasks. So let's have a look at some of the features engineering tools.

FeatureTools

Featuretools is a framework to perform automated feature engineering. It excels at transforming temporal and relational datasets into feature matrices for machine learning. Featuretools integrates with the machine learning pipeline-building tools you already have. In a fraction of the time it would take to do it manually, you can load in pandas dataframes and automatically construct significant features.

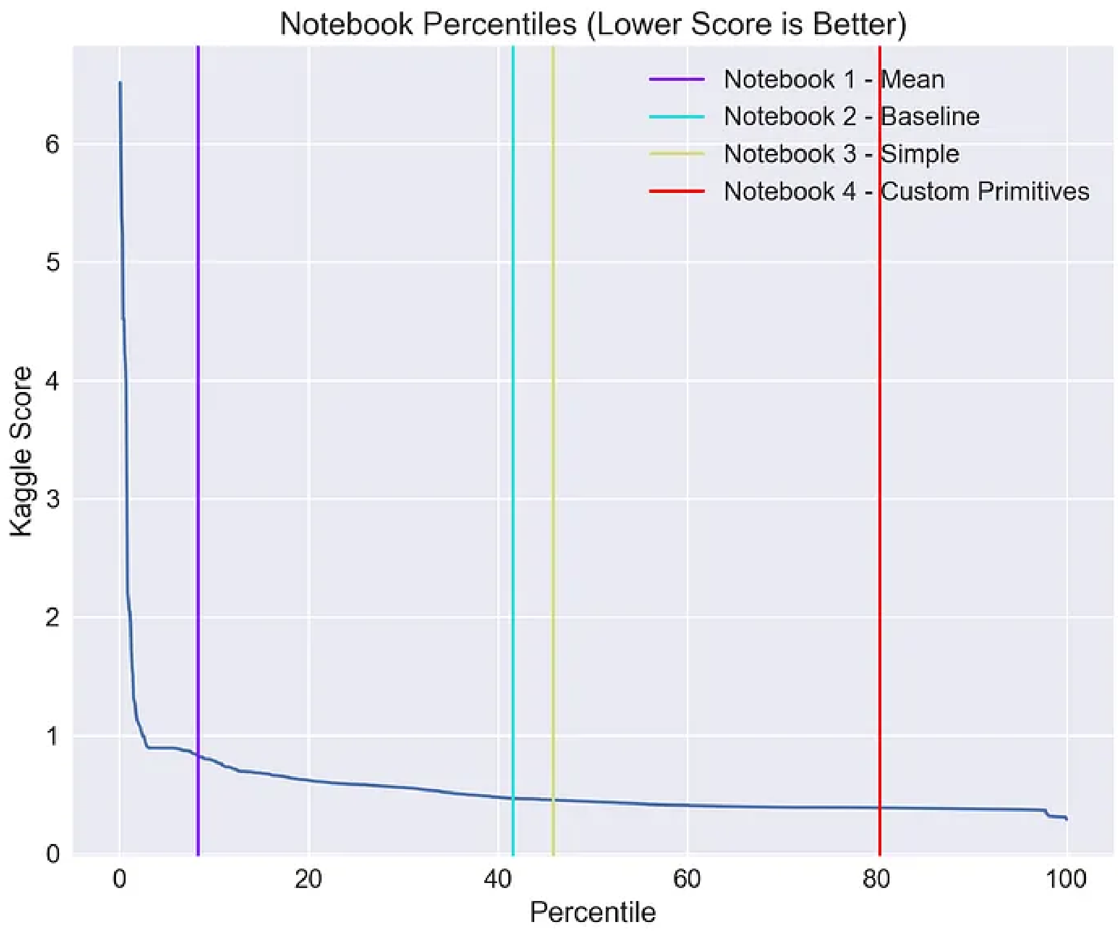


Image Source

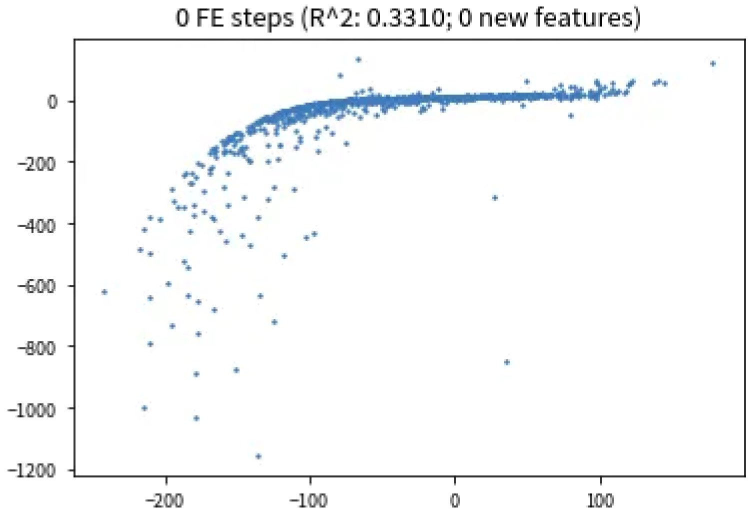
FeatureTools Summary

- Easy to get started, good documentation and community support
- It helps you construct meaningful features for machine learning and predictive modelling by combining your raw data with what you know about your data.
- It provides APIs to verify that only legitimate data is utilised for calculations, preventing label leakage in your feature vectors.
- Featuretools includes a low-level function library that may be layered to generate features.
- Its AutoML library(EvalML) helps you build, optimize, and evaluate machine learning pipelines.
- Good at handling relational databases.

Learn More About FeatureTools.

AutoFeat

AutoFeat helps to perform Linear Prediction Models with Automated Feature Engineering and Selection. AutoFeat allows you to select the units of the input variables in order to avoid the construction of physically nonsensical features.



Source

AutoFeat Summary

- AutoFeat can easily handle categorical features with One hot encoding.
- The AutoFeatRegressor and AutoFeatClassifier models in this package have a similar interface to scikit-learn models
- General purpose automated feature engineering which is Not good at handling relational data.
- It is useful in logistical data

Learn More About AutoFeat.

TsFresh

tsfresh is a python package. It calculates a huge number of time series characteristics, or features, automatically. In addition, the package includes methods for assessing the explanatory power and significance of such traits in regression and classification tasks.

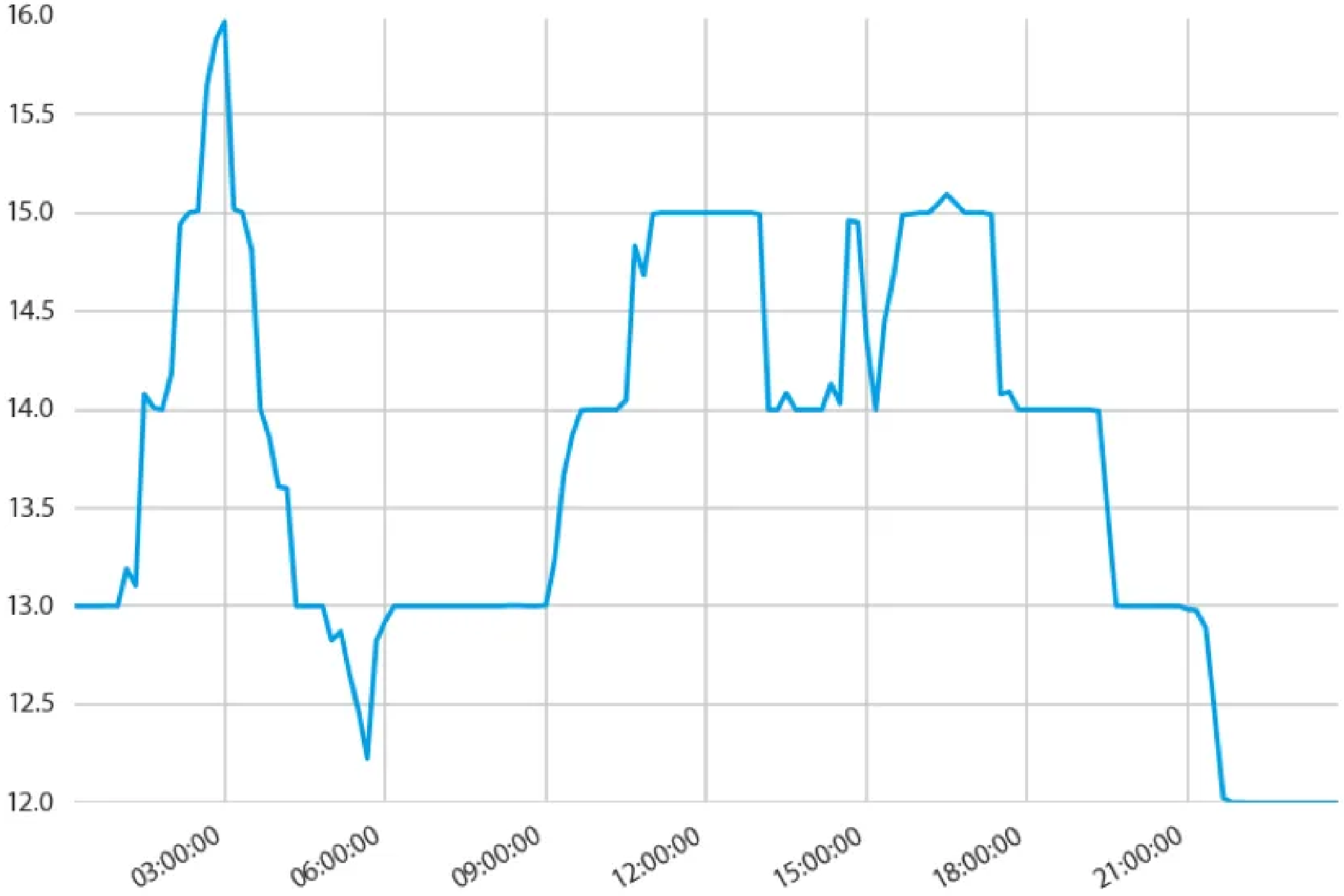


Image Source

TsFresh Summary

- It is Best open source python tool available for time series classification and regression.
- It helps to extract things such as the number of peaks, average value, maximum value, time reversal symmetry statistic, etc.
- It can be integrated with FeatureTools.

Learn More About TsFresh .

OneBM

OneBM interacts directly with a database's raw tables. It slowly joins the tables, taking different paths on the relational tree. It recognises simple data types (numerical or categorical) and complicated data types (set of numbers, set of categories, sequences, time series, and texts) in the joint results and applies pre-defined feature engineering approaches to the supplied types.

- Both relational and non-relational data are supported.

- When compared to FeatureTools, it generates both simple and complicated features.
- It was put to the test in Kaggle competitions, and it outperformed state-of-the-art models.

Learn More About OneBM

ExploreKit

Based on the idea that extremely informative features are typically the consequence of manipulating basic ones, ExploreKit identifies common operators to alter each feature independently or combine multiple of them. Instead of running feature selection on all developed features, which can be quite huge, meta learning is used to rank candidate features.

Learn More About ExploreKit

Comparison

Conclusion

Feature engineering is the development of new data features from raw data. With this technique, engineers analyze the raw data and potential information in order to extract a new or more valuable set of features. Feature engineering can be seen as a generalization of mathematical optimization that allows for better analysis. Hope you learned about feature engineering, its techniques and tools used by engineers. If you have any doubt regarding the article you can drop a comment.

Stay Safe and Happy Experimenting !!

References and Recommend Reading :

- https://www.omnisci.com/technical-glossary/feature-engineering

- https://acuvate.com/blog/the-what-why-and-how-of-feature-engineering/

- https://neptune.ai/blog/feature-engineering-tools
- https://www.analyticsvidhya.com/blog/2021/03/step-by-step-process-of-feature-engineering-for-machine-learning-algorithms-in-data-science/
- https://analyticsindiamag.com/guide-to-automatic-feature-engineering-using-autofeat/
- https://medium.com/analytics-vidhya/automated-feature-engineering-tools-44d00be56e3a
- https://neptune.ai/blog/ml-from-research-to-production

Feature Engineering Machine Learning Data Science Mlops Tools

Follow

Written by Harshil Patel

717 Followers · Writer for Towards Data Science

Software Developer and Technical Writer.

More from Harshil Patel and Towards Data Science

Harshil Patel in Level Up Coding

Hacking the Dino Game from Google Chrome

When there is no internet connection available, Google Chrome web browser on Windows and macOS (most likely on Linux too) shows up a page…

3 min read · Jan 12, 2020

689 18

Sheila Teo in Towards Data Science

How I Won Singapore’s GPT-4 Prompt Engineering Competition

A deep dive into the strategies I learned for harnessing the power of Large Language Models

✦ · 24 min read · 5 days ago

2.5K 33

Mariya Mansurova in Towards Data Science

Can LLMs Replace Data Analysts? Building An LLM-Powered Analyst

Part 1: empowering ChatGPT with tools

19 min read · Dec 11, 2023

1.2K 12

Harshil Patel in Towards Data Science

An Overview of QuickSort Algorithm

Sorting is the process of organizing elements in a structured manner. Quicksort is one of the most popular sorting algorithms that uses…

9 min read · Mar 10, 2022

148 2

See all from Harshil Patel

See all from Towards Data Science

Recommended from Medium

Everton Gomede, PhD

The Significance of Train-Validation-Test Split in Machine Learning

Introduction

8 min read · Aug 28, 2023

23 1

Hasan Hüseyin Coşgun

Which data scaling technique should I use ?

Data scaling reduces bias impact in Machine Learning. In Article compares StandardScaler, MinMaxScaler, RobustScaler.

4 min read · Aug 5, 2023

33

Lists

Predictive Modeling w/ Python

20 stories · 746 saves

Practical Guides to Machine Learning

10 stories · 865 saves

Natural Language Processing

1053 stories · 529 saves

data science and AI

38 stories · 31 saves

Diborah Kiptoon

Feature Selection in Machine Learning

Understanding Various Methods and Techniques of Feature Selection

12 min read · Aug 18, 2023

67 1

Mochamad Kautzar Ichramsyah in CodeX

Automate the exploratory data analysis (EDA) to understand the data faster and easier

What is EDA?

11 min read · Jul 11, 2023

2.3K 23

Virat Patel

I applied to 230 Data science jobs during last 2 months and this is what I’ve found.

A little bit about myself: I have been working as a Data Analyst for a little over 2 years. Additionally, for the past year, I have been…

✦ · 3 min read · Aug 11, 2023

2.8K 54

Jones ntongana

Understanding Interaction and Polynomial Features in PySpark: A Simple Guide

Introduction

3 min read · Sep 1, 2023

See more recommendations