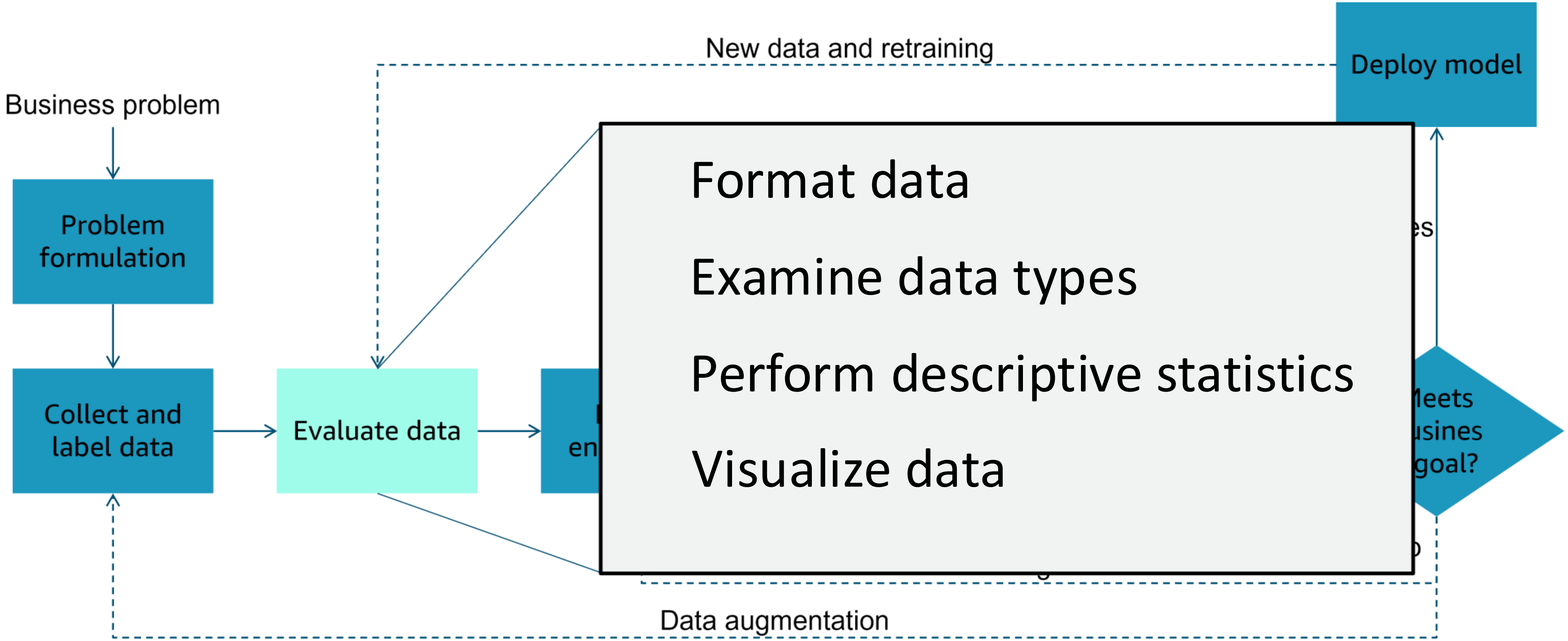
Feature Engineering

Outline

- Data Evaluation
- Feature engineering

Data Evaluation

ML Pipeline: Evaluate Data



Format data

Examine data types

Perform descriptive statistics

Visualize data

Source: Amazon Web Services

Understand Your Data

"Customer:ABC,DateOfTransation:10/5,Vendor:Store1,ChargeAmount:10.99,WasThisFruad:No…"

Customer	Date of Transaction	Vendor	Charge Amount	Was This Fraud?
ABC	10/5	Store 1	10.99	No
DEF	10/5	Store 2	99.99	Yes
GHI	10/5	Store 2	15.00	No
JKL	10/6	Store 2	99.99	?
MNO	10/6	Store 1	99.99	Yes

Source: Amazon Web Services

Load Data

- Reformats data into tabular representation (DataFrame)
    - Rows
    - Columns
- Converts common formats like comma-separated values (csv), text file (txt), JavaScript Object Notation (JSON), Excel, and others

```
import pandas as pd
url = "https://somewhere.com/winequality-red.csv"
df_wine = pd.read_csv(url,';')
```

Source: Amazon Web Services

Load Data: Pandas DataFrame

Number of instances

```
df_wine.shape
```

(1599, 12)

Number of attributes

```
df_wine.head(5)
```

Columns/Attributes

Rows/Instances

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5
1	7.8	0.88	0.00	2.6	0.098	25.0	67.0	0.9968	3.20	0.68	9.8	5
2	7.8	0.76	0.04	2.3	0.092	15.0	54.0	0.9970	3.26	0.65	9.8	5
3	11.2	0.28	0.56	1.9	0.075	17.0	60.0	0.9980	3.16	0.58	9.8	6
4	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5

Source: Amazon Web Services

Index and Column Names

```
df_wine.columns
```

```
Index(['fixed acidity', 'volatile acidity', 'citric acid', 'residual
sugar’, 'chlorides', 'free sulfur dioxide', 'total sulfur dioxide',
'density’, 'pH', 'sulphates', 'alcohol', 'quality’],
dtype='object')
```

```
df_wine.index
```

```
RangeIndex(start=0, stop=1599, step=1)
```

Source: Amazon Web Services

Data Type

df_wine.dtypes()

```
quality                         int64
fixed acidity                   float64
volatile acidity                float64
citric acid                     float64
residual sugar                  float64
chlorides                       float64
free sulfur dioxide             float64
total sulfur dioxide            float64
density                         float64
pH                              float64
sulphates                       float64
alcohol                         float64
dtype: object
```

```
df_data['col'] = df_data['col'].astype('int')
```

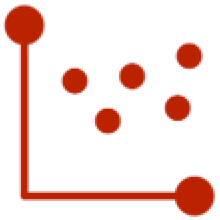
df_wine.info()

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1597 entries,
0 to 1598
Data columns (total 12 columns):
quality                   1597 non-null int64
fixed acidity             1597 non-null float64
volatile acidity          1597 non-null float64
citric acid               1597 non-null float64
residual sugar            1597 non-null float64
chlorides                 1597 non-null float64
free sulfur dioxide       1597 non-null float64
total sulfur dioxide      1597 non-null float64
density                   1597 non-null float64
pH                        1597 non-null float64
sulphates                 1597 non-null float64
alcohol                   1597 non-null float64
dtypes: float64(11), int64(1)
memory usage: 162.2 KB
```

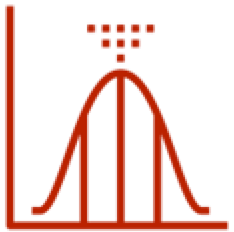
Descriptive Statistics

Use descriptive statistics to **gain insights** into your data before you clean the data:

Overall statistics



Multivariate statistics



Attribute statistics

Statistical Characteristics

```
df_wine.describe()
```

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	pH	sulphates	alcohol	quality
count	1599.00	1599.00	1599.00	1599.00	1599.00	1599.00	1599.00	1599.00	1599.00	1599.00	1599.00
mean	8.32	0.53	0.27	2.54	0.09	15.87	46.47	3.31	0.66	10.42	5.64
std	1.74	0.18	0.19	1.41	0.05	10.46	32.90	0.15	0.17	1.07	0.81
min	4.60	0.12	0.00	0.90	0.01	1.00	6.00	2.74	0.33	8.40	3.00
25%	7.10	0.39	0.09	1.90	0.07	7.00	22.00	3.21	0.55	9.50	5.00
50%	7.90	0.52	0.26	2.20	0.08	14.00	38.00	3.31	0.62	10.20	6.00
75%	9.20	0.64	0.42	2.60	0.09	21.00	62.00	3.40	0.73	11.10	6.00
max	15.90	1.58	1.00	15.50	0.61	72.00	289.00	4.01	2.00	14.90	8.00

Categorical Statistics

Identify frequency of values and class imbalance

`df_car.head(5)`

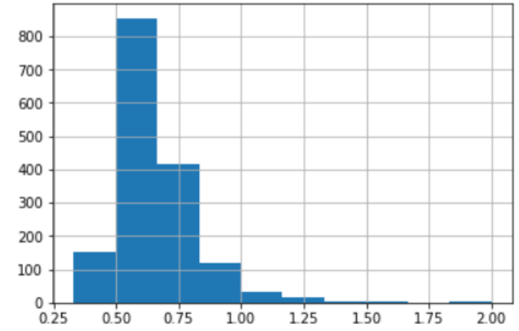
	buying	maint	doors	persons	lug_boot	safety	class
0	vhigh	vhigh	2	2	small	low	unacc
1	vhigh	vhigh	2	2	small	med	unacc
2	vhigh	vhigh	2	2	small	high	unacc
3	vhigh	vhigh	2	2	med	low	unacc
4	vhigh	vhigh	2	2	med	med	unacc

`df_car.describe()`

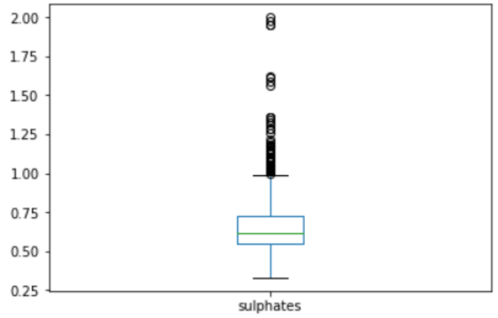
	buying	maint	doors	persons	lug_boot	safety	class
count	1728	1728	1728	1728	1728	1728	1728
unique	4	4	4	3	3	3	4
top	low	low	2	2	big	low	unacc
freq	432	432	432	576	576	576	1210

Plotting Attribute & Multivariate Statistics

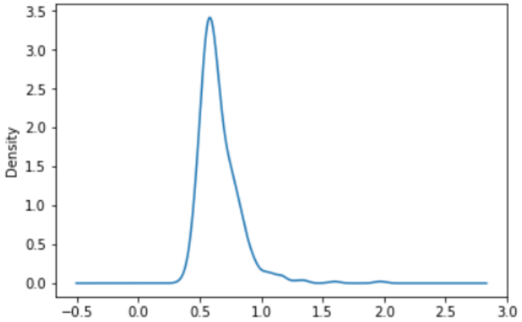
```
df_wine[‘sulphates’].hist(bins=
```



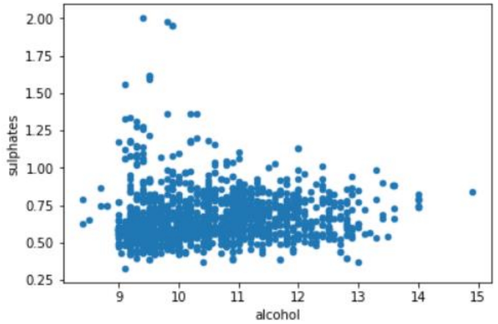

```
df_wine[‘sulphates’].plot.box()
```




```
df_wine[‘sulphates'].plot.kde()
```

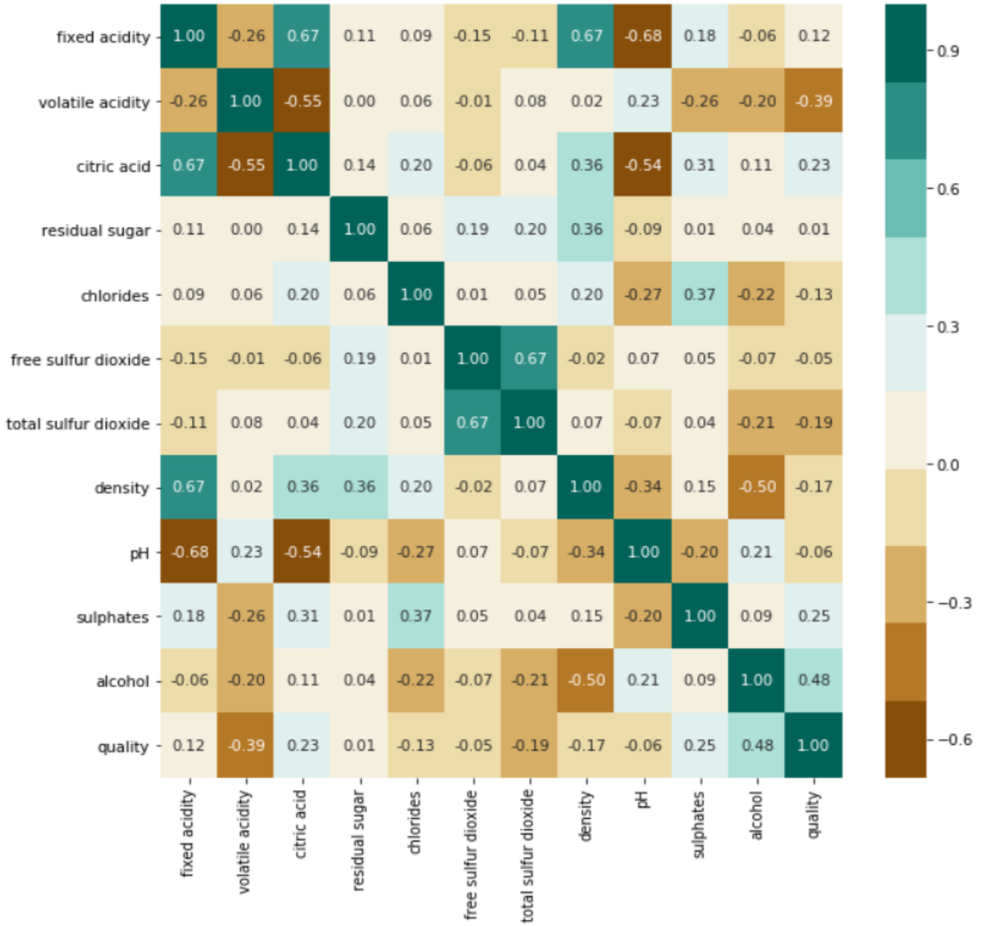



```
df_wine.plot.scatter(
          x='alcohol', y='sulphates')
```



Correlation Matrix Heat Map

```
import seaborn as sns

correlations = df_wine.corr()
fig, ax = plt.subplots(figsize=(10, 10))
colormap = sns.color_palette("BrBG", 10)

sns.heatmap(correlations,  cmap=colormap,
                    annot=True, fmt=".2f")

ax.set_yticklabels(colum_names);
plt.show()
```



Key takeaways

- The first step in evaluating data is to make sure that it's in the right format.
- Pandas is a popular Python library for working with data.
- Use descriptive statistics to learn about the dataset.
- Create visualizations with pandas to examine the dataset in more detail.

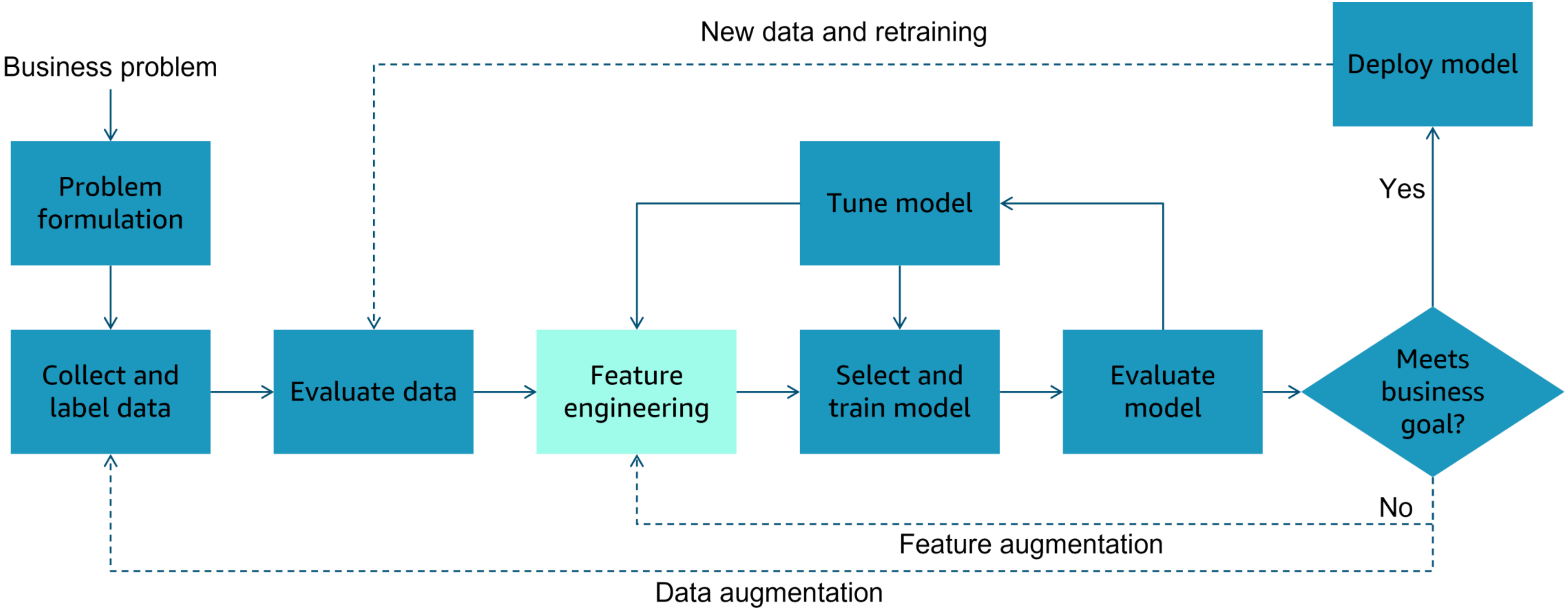
Feature Engineering

Feature Engineering (a definition)

Feature engineering is a preprocessing step in supervised machine learning and statistical modeling[1] which transforms raw data into a more effective set of inputs. Each input comprises several attributes, known as features. By providing models with relevant information, feature engineering significantly enhances their predictive accuracy and decision-making capability.[2][3][4]

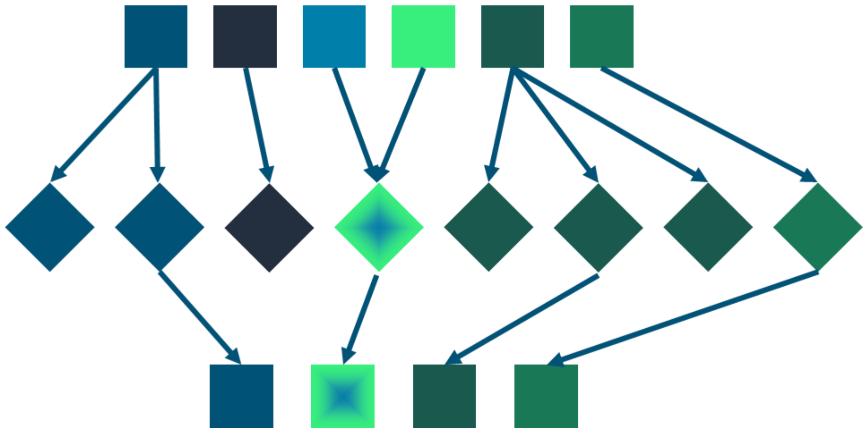
Wikipedia contributors. (2024, August 19). Feature engineering. In *Wikipedia, The Free Encyclopedia*. Retrieved 20:07, October 13, 2024, from https://en.wikipedia.org/w/index.php?title=Feature_engineering&oldid=1241120615

ML Workflow: Feature Engineering



Feature Selection & Extraction

Feature Extraction

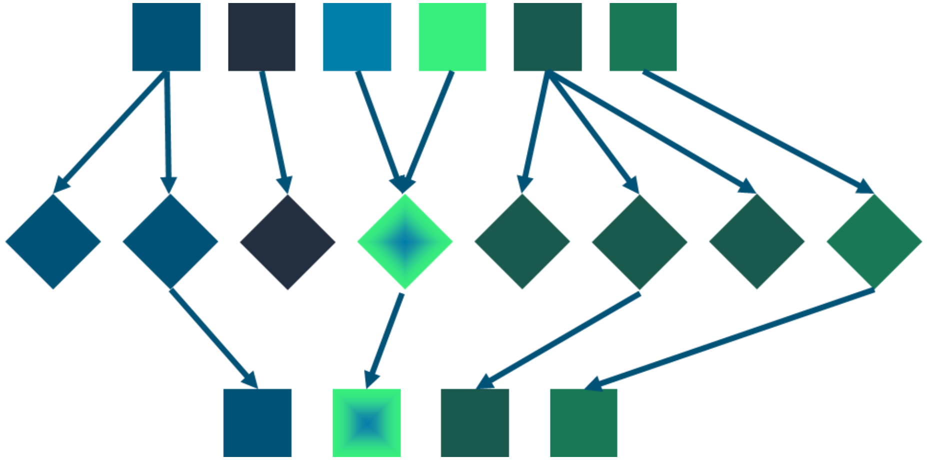


To build up **valuable information** from raw data by **reformatting**, **combining**, and **transforming** primary features into new ones.

Feature Selection

To **prevent** either **redundancy** or **irrelevance** in the existing features, or to get a limited number of features to prevent **overfitting**.

Feature Extraction: Data Handling



- Wrong formats
- Invalid values
- Misspelling
- Duplicates
- Encode categories (text -> numeric)
- Consistency
- Remove outliers
- Reassign outliers
- Rescale
- Transformation
- Consistency
- Combine data
- Split data into multiple columns

Encoding Data

Categorical data is non-numeric data.

Categorical data must be converted (encoded) to a numeric scale.

Tools such as Scikit-Learn and Pandas can be used to encode your categorical data after you make sure that it is all uniform.

Maintenance Costs	Encoding
Low	1
Medium	2
High	3
Very High	4

Encoding Non-ordinal Data

If data is non-ordinal, the encoded values also must be non-ordinal.

Non-ordinal data might need to be broken into multiple categories.

...	Color
...	Red
...	Blue
...	Green
...	Blue
	Green

...	Red	Blue	Green
...	1	0	0
...	0	1	0
...	0	0	1
...	0	1	0
...	0	0	1

Cleaning Data

Types of data to clean:

Type	Example	Action
Variations in strings	Med. vs. Medium	Convert to standard text
Variations in scale	Number of doors vs. number of cars purchased	Normalize to a common scale
Columns with multiple data items	Safe high-maintenance	Parse into multiple columns
Missing data	Missing columns of data	Delete rows or impute data
Outliers	Various	

Finding Missing Data

- Missing data makes it difficult to interpret relationships
- Causes of missing data:
  - Undefined values
  - Data collection errors
  - Data cleaning errors
- Example pandas code to find missing data:

```
df.isnull().sum() #count missing values for each column
df.isnull().sum(axis=1) #count missing values for each row
```

Dropping Missing Values

Drop missing data with pandas

- dropna function to drop rows

```
df.dropna()
```

- dropna function to drop columns with null values

```
df.dropna (axis=1)
```

- dropna function to drop a subset

```
df.dropna(subset=[“buying”])
```

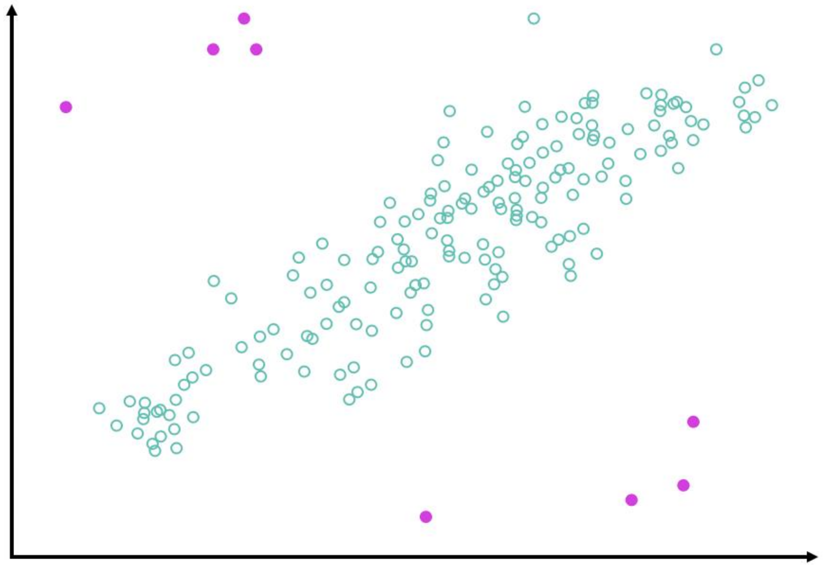
Imputing Missing Values

- First, determine why the data is missing
- Two ways to impute missing data:
  - Univariate: Adding data for a single row of missing data
  - Multivariate: Adding data for multiple rows of missing data

```
from sklearn.preprocessing import Imputer

import numpy as np

Arr = np.array([[5,3,2,2],[3,None,1,9],[5,2,7,None]])

imputer = Imputer(strategy='mean’)

imp = imputer.fit(arr)

imputer.transform(arr)
```

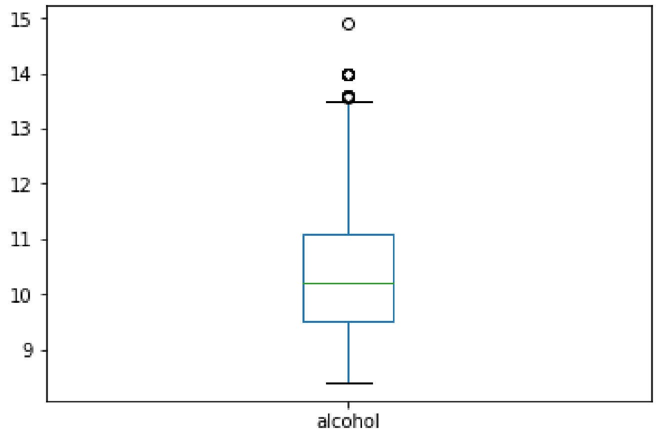
Outliers

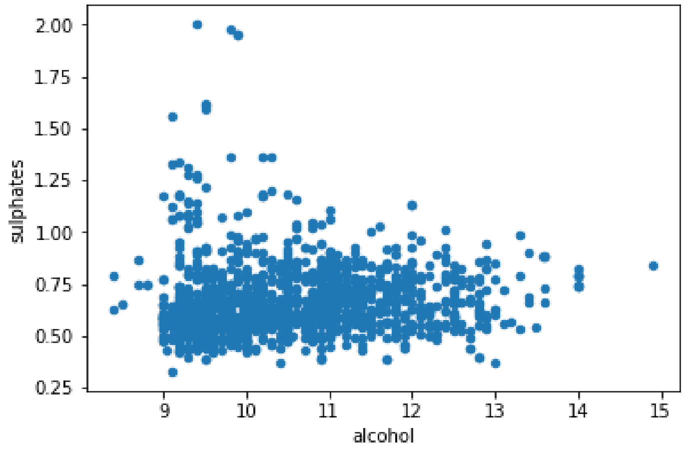
- Outliers can:
  - Provide a broader picture of the data
  - Make accurate predictions difficult
  - Indicate the need for more columns
- Types of outliers
  - Univariate: Abnormal values for a single variable
  - Multivariate: Abnormal values for a combination of two or more variables



Finding Outliers

- Box plots show variation and distance from the mean
  - Example shows a box plot for the amount of alcohol in a collection of wines
- Scatter plots can also show outliers
  - A scatter plot shows the relationship between alcohol and sulphates in a collection of wines





Dealing with Outliers

Delete the outlier	Transform the outlier	Impute a new value for the outlier
Outlier is based on an artificial error.	Reduces the variation that the extreme outlier value causes and the outlier's influence on the dataset.	You might use the mean of the feature, for instance, and impute that value to replace the outlier value.

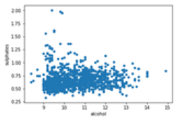
Feature Selection: Filter Methods

Measures:

- Pearson’s correlation coefficient
- Linear discriminant analysis (LDA)
- Analysis of variance (ANOVA)
- Chi-square

All features

Statistics and correlation

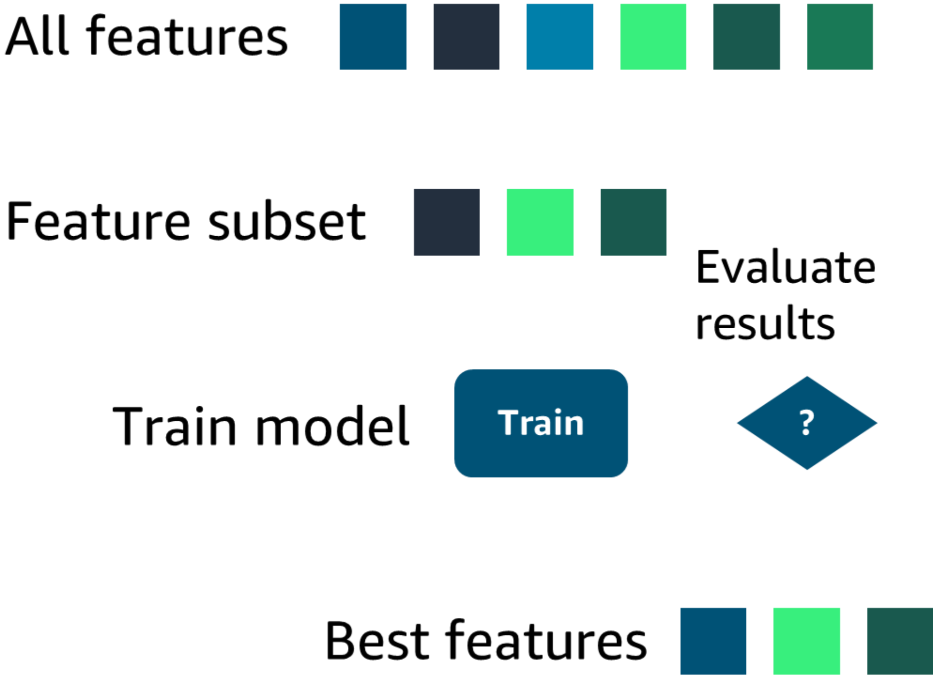


Best features

Feature Selection: Wrapper

Methods :

- Forward selection
- Backward selection



Key takeaways

- Feature engineering involves :
  - Selection
  - Extraction
- Pre-processing gives you better data
- Two categories for preprocessing:
  - Converting categorical data
  - Cleaning up dirty data
- Use encoding to convert categorical data
- Various types of dirty data:
  - Missing data
  - Outliers
- Develop a strategy for cleaning dirty data
  - Replace or delete rows with missing data
  - Delete, transform, or impute new values for outliers