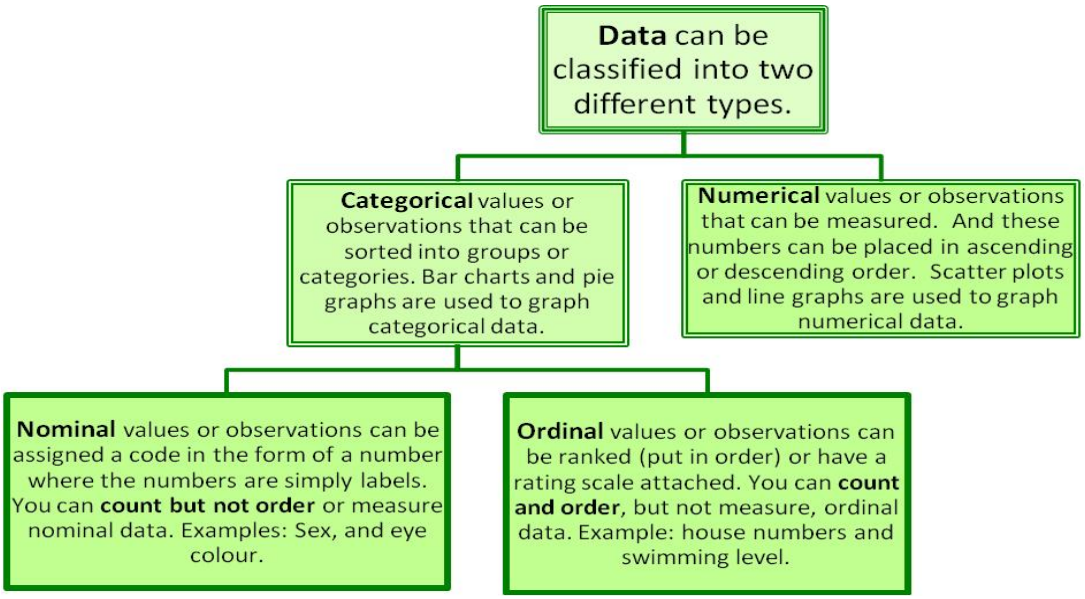
1- Types of features



Dealing with numeric data in machine learning is often easier than categorical data, given that we do not have to deal with additional complexities of the semantics pertaining to each category value in any data attribute which is of a categorical type.

Many machine learning algorithms cannot operate on categorical data directly. Here, data must be transformed into a numerical form.

2- Conversion of categorical features into numerical features

- Label encoder

ID	Country	Population
1	Japan	127185332
2	U.S	326766748
3	India	1354051854
4	China	1415045928
5	U.S	326766748
6	India	1354051854

ID	Country	Population
1	0	127185332
2	1	326766748
3	2	1354051854
4	3	1415045928
5	1	326766748
6	2	1354051854

Depending on the data, label encoding introduces a new problem. For example, we have encoded a set of country names into numerical data. This is actually categorical data and there is no relation, of any kind, between the rows. The problem here is since there are different numbers in the same column, the model will misunderstand the data to be in some kind of order, 0 < 1 <2.

The model may derive a correlation like as the country number increases the population increases, but this clearly may not be the scenario in some other data or the prediction set. To overcome this problem, we use One-Hot Encoder.

- One-Hot Encoder

One-hot encoding takes a categorical column, which has been label encoded, and then splits it into multiple columns. The numbers are replaced by 1s and 0s, depending on which column has what value. In our example, we get four columns, one for each country — Japan, U.S, India, and China.

ID	Country_Japan	Country_U.S	Country_India	Country_China	Population
1	1	0	0	0	127185332
2	0	1	0	0	326766748
3	0	0	1	0	1354051854
4	0	0	0	1	1415045928
5	0	1	0	0	326766748
6	0	0	1	0	1354051854

One-hot encoding is fine for categorical variables with a few possible values. But it has two main drawbacks:

1. For high-cardinality variables — those with many unique categories — the dimensionality of the transformed vector becomes unmanageable.
2. The mapping is completely uninformed: similar categories are not placed closer to each other in embedding space.

- Embeddings

Neural network embeddings overcome the two limitations of One-hot encoding.

3- Dimensionality reduction

It allows summarizing a set of correlated variables with a smaller set of variables that collectively explain most of the variability in the original set. Basically, we drop the least important features.

Principal Component Analysis (PCA) is the process by which principal components are calculated and used to analyze and understand the data. PCA is an unsupervised learning approach that is used for dimensionality reduction, feature extraction, and data visualization.

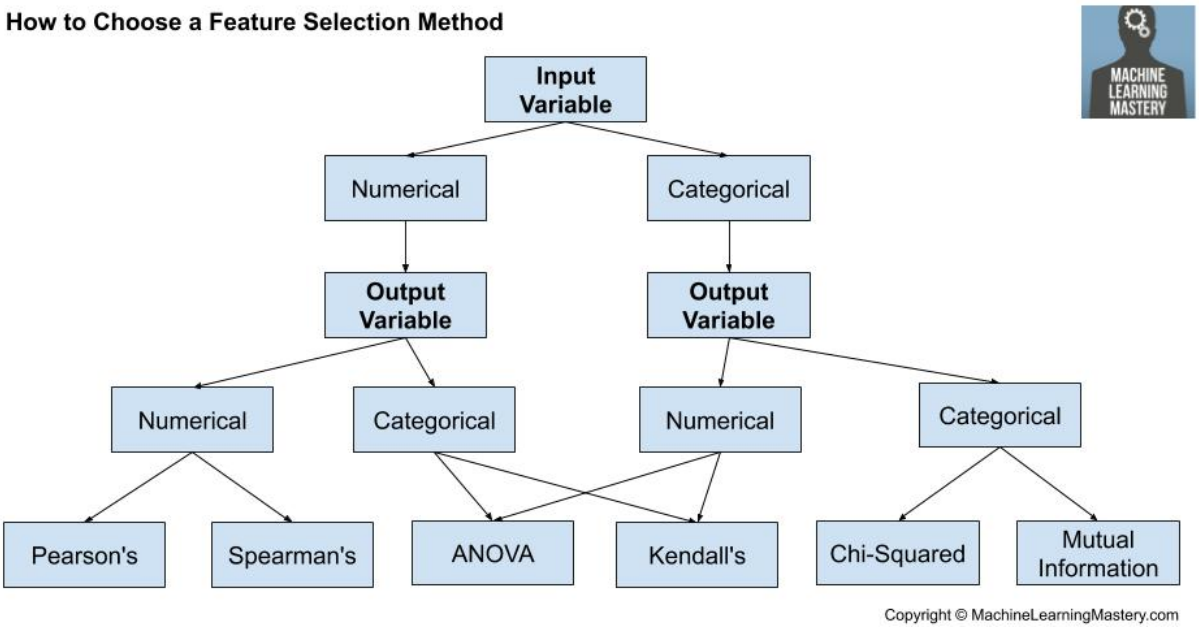
Scale variables is important to perform PCA. The variables after performing PCA are independent.

4- Feature selection

Visual exploration of relationship between variables

- Continuous vs Continuous ---- Scatter Plot
- Categorical vs Continuous---- Side-by-side Box Plot
- Continuous vs Categorical--- Grouped Histograms or Side-by-side Box Plot
- Categorical vs Categorical---- Grouped Bar Plots

Statistical measurement of relationship strength between variables



ANOVA (Analysis of Variance)

ANOVA is conducted to check the correlation between a categorical and a numerical variable.

For example, does bordering a river have a relationship with the price of a house?

Assumption H0: There is no relation between the given variables.

F = variance between categorical variables / variance within categorical variables

P-value = F / 100

P is the probability that the variables are independent (reject H0 if P < 0.05)
The larger the ratio, the more likely the variables have no relationship.

Chi-Square test

The Chi-Square test is conducted to check the correlation between two categorical variables.
For example, is there a relationship between gender and preference for pets (cats or dogs)?
Assumption H0: The two columns are not related to each other.

	Cat	Dog	
Men	207	282	489
Women	231	242	473
	438	524	962

	Cat	Dog	
Men	$\frac{489 \times 438}{962}$	$\frac{489 \times 524}{962}$	489
Women	$\frac{473 \times 438}{962}$	$\frac{473 \times 524}{962}$	473

	Cat	Dog	
Men	222,64	266,36	489
Women	215,36	257,64	473
	438	524	962

Subtract expected from observed, square it, then divide by expected:

In other words, use formula $\frac{(O-E)^2}{E}$ where

- O = **Observed** (actual) value
- E = **Expected** value

	Cat	Dog	
Men	$\frac{(207-222,64)^2}{222,64}$	$\frac{(282-266,36)^2}{266,36}$	489
Women	$\frac{(231-215,36)^2}{215,36}$	$\frac{(242-257,64)^2}{257,64}$	473
	438	524	962

	Cat	Dog	
Men	1,099	0,918	489
Women	1,136	0,949	473
	438	524	962

P-value = 1,099 + 0,918 + 1,136 + 0,949 = 4,102

P is the probability that the variables are independent (reject H0 if P < 0.05).
The larger the ratio, the more likely the variables have no relationship.

Mutual information

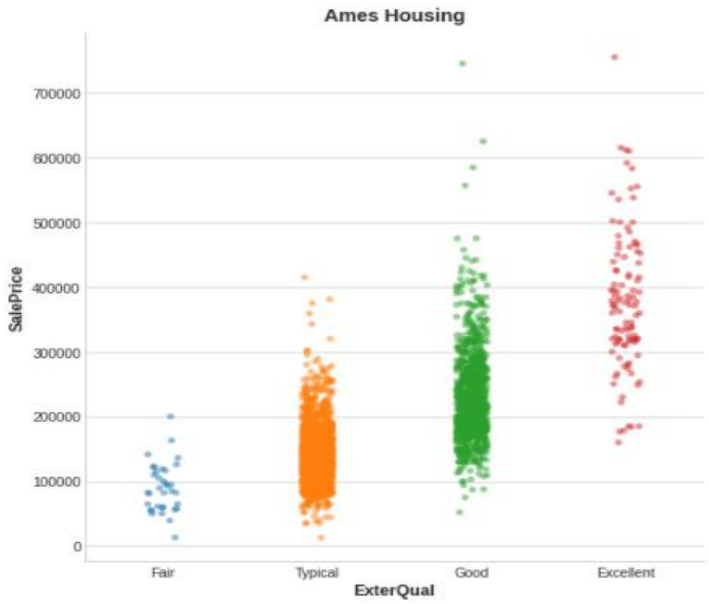
The main advantage of mutual information over correlation is that it can detect any kind of relationship, while correlation only detects linear relationships. Mutual information is:

- easy to use and interpret,
- computationally efficient,
- theoretically well-founded,
- resistant to overfitting.

Mutual information describes relationships in terms of *uncertainty*. The mutual information between two quantities measures the extent to which knowledge of one quantity reduces uncertainty about the other. If you knew the value of a feature, how much confident would you be about the target?

Here's below an example from the *Ames Housing* data. This figure shows the relationship between the exterior quality of a house and its price. Each point represents a house.

Uncertainty is measured using a quantity from information theory known as *Entropy*. The entropy of a variable means: "how many yes-or-no questions you need to describe an occurrence of that variable, on average." The more questions you ask, the more uncertain you are about the variable. Mutual information is how many questions you expect the feature to answer about the target.



Knowing the exterior quality of a house reduces uncertainty about its sale price.

5- Create new features

Tips to discover new features:

- Understand the features. Refer to the documentation of the dataset, if available.
- Research the problem domain to acquire domain knowledge. For example, you can do some research on real estate if your problem is predicting house prices. Wikipedia can be a good starting point, but books and journal articles will often have the best information.
- Study previous work. Solution write-ups from past Kaggle competitions are a great resource.
- Use data visualization. Visualization can reveal pathologies in the distribution of a feature or complicated relationships that could be simplified.
- Linear models naturally learn sums and differences, but can't learn anything more complex.
- Ratios seem to be difficult for most models to learn. Ratio combinations often lead to some easy performance gains.
- Tree models can approximate almost any combination of features, but when a combination is important, they can benefit from having it explicitly created, especially with limited data.
- Counts are especially helpful for tree models, since these models don't have a natural way of aggregating information across many features at once.

Mathematical Transforms:

Relationships among numerical features are often expressed through mathematical formulas.

For example, in the *Automobile* dataset, there are features describing a car's engine. The "stroke ratio", for instance, is a measure of how efficient an engine is versus how performant.

Counts:

In the *Traffic Accidents* dataset, there are features indicating whether a roadway object was near the accident. We can count the total number of roadways features nearby using the sum method.

Building-Up and Breaking-Down Features

Complex strings can usefully be broken into simpler pieces. Some common examples are as follows:

- ID numbers: '123-45-6789'
- Phone numbers: '(999) 555-0123'
- Street addresses: '8241 Kaggle Ln., Goose City, NV'
- Internet address: 'http://www.kaggle.com'
- Product codes: '0 36000 29145 2'

Features like these will often have a structure that you can make use of. US phone numbers, for instance, have an area code (the '(999)' part) that tells you the location of the caller.

You could also join simple features into a composed feature if you had reason to believe there was some interaction in the combination. For example, as follows:

Make: Audi, Body style: Sedan, Make and style: Audi_Sedan

Group Transforms

Group transforms aggregate information across multiple rows grouped by some category. With a group transform, you can create features like: "the average income of a person's state of residence", or "the proportion of movies released on a weekday, by genre". If you discovered a category interaction, a group transform over that category could be something good to investigate.

A group transform combines two features using an aggregation function: a categorical feature that provides the grouping and another feature you wish to aggregate. For example, for an average income by state, you would choose “State” for the grouping feature, the Mean for the aggregation function, and “Income” for the aggregated feature.