SUMMARY:

Expected input and output data format for various feature transformation techniques.

Technique	Input	Output
Binary Indicator	Categorical or Numeric	Binary (0/1)
One-Hot Encoding	Categorical	Binary (0/1)
Functional Transformation	Numeric	Numeric
Interaction Terms	Numeric	Numeric
Binning	Numeric	Categorical
Scaling	Numeric	Numeric

INTRODUCTION TO EXPLORATORY DATA ANALYSIS

Assume we're at the point where we have a representative sample of users & have a structured dataset that includes features, and if appropriate, a label.

Approach our data investigation phase with 2 main goals:

1. Ensure we have high-quality data
   - ↳ Bad quality data: missing values, outliers, etc.
2. Delievering insights so we can be knowledgable about the problem.
   - ↳ EXPLORATORY DATA ANALYSIS (EDA):
     - Explore your data by asking several key questions about your data, which are oriented towards supervised learning.

Made w/ direct utility to the model-vetting process

- ↳ How is the data distributed? ⟹ informs about outliers & skews we should address
- ↳ What features are redundant? ⟹ informs what features we may or may not cut.
- ↳ How do different features correlate with our label? ⟹ informs about features to keep & offers high expectations on how modeling might be working.

GOAL: We want to end up with well distributed independent features.

When examining / working with data:

STEP 1: Look at it

- ↳ You're given a flat text file
- ↳ Before loading it into python, check for a few things!

USING PANDAS TO INSPECT YOUR DATA

When performing EDA our main goals is to check for data quality issues. → missing values, outliers

↳ Pandas method: pd.dataframe. describe

EX)

```
In [27]: describe_vars = ['isbuyer','buy_freq','expected_time_buy','uniq_urls','num_checkins','y_buy']
         df[describe_vars].describe()
```

← filtered to a limited set of columns

Out[27]:

	isbuyer	buy_freq	expected_time_buy	uniq_urls	num_checkins	y_buy
count	54584.000000	2327.000000	54584.000000	54584.000000	54584.000000	54584.000000
mean	0.042632	1.240653	-0.198040	86.569343	720.657592	0.004635
std	0.202027	0.782228	4.997792	61.969765	1275.727306	0.067924
min	0.000000	1.000000	-181.923800	-1.000000	1.000000	0.000000
25%	0.000000	1.000000	0.000000	30.000000	127.000000	0.000000
50%	0.000000	1.000000	0.000000	75.000000	319.000000	0.000000
75%	0.000000	1.000000	0.000000	155.000000	802.000000	0.000000
max	1.000000	15.000000	84.285710	206.000000	37091.000000	1.000000

← binary label

The count of records that have non-null values

Tells us over 95% of the column values are missing.

2 columns here have negative values. It can be an encoding value error.

We'd usually do some pre-processing to address this particular concern

STANDARD DISTRIBUTION STATISTICS

Examining this, the average is incredibly low

GENERALLY, if your binary label is 1 less than 1% of the time ⇒ CLASS IMBALANCE PROBLEM

mean is greater than the median ⇒ HIGHLY SKEWED COLUMN

⇓

RELATED TO HAVING OUTLIERS

VISUALIZE YOUR DATA: UNVARIANT PLOTTING

Plotting your data gives a lot of information that summary statistics don't.

Using plots helps us understand 2 basic things:

① How is a given feature distributed? [UNIVARIANT INSIGHT]
  ↳ Tells us the range of the data, central tendency, skew, and variance.
  ↳ knowing this will tell us what preparation techniques we might need to use.

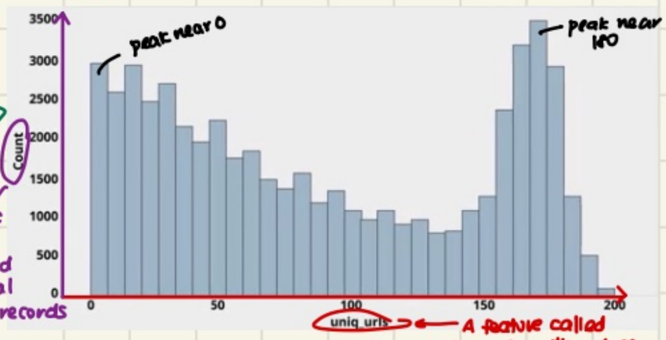
② Know how two or more features are distributed together.
  ↳ Tells if 2 features are correlated or redundant.

PLOTS TO UNDERSTAND UNIVARIANT DISTRIBUTIONS

① HISTOGRAM: x axis == range of data (specific features)
y axis == how much of the data is present in that range.

↳ Perfect to understand if there's a heavy skew
↳ Works for both numeric & categorical data
↳ Using SEABORN and MATPLOTLIB to produce these samples.

EX) This shows that we have a multi-modal distribution



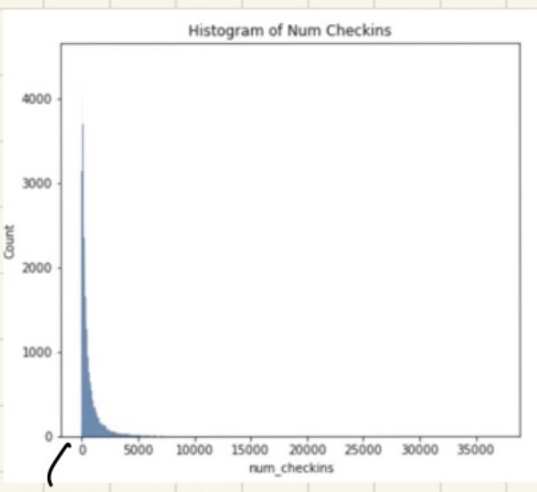
Sum of the bar should add up the total number of records

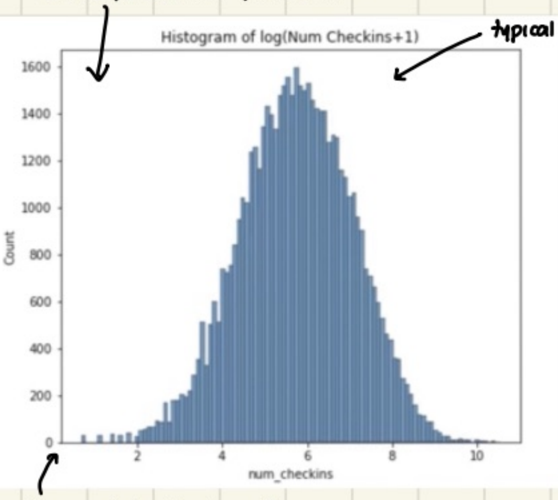
A feature called uniq_urls with values from 0–200.

would yield better performance

typical bell curve for normal distribution

EX)





Most data is in the first 5% or 10%

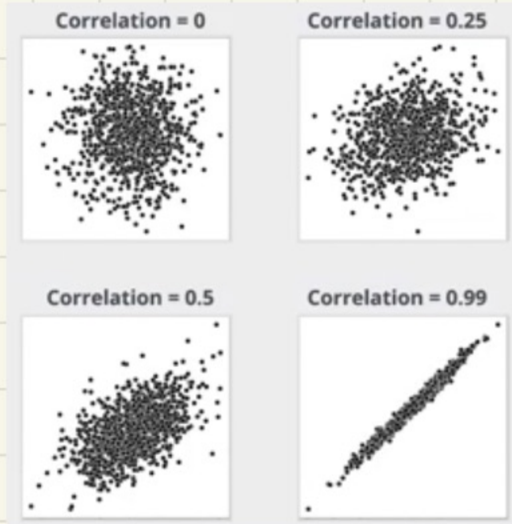
when we take the log of the feature

• This is a good example of a feature with which we may want to form a functional transformation w/ the function here being the natural log.

• When a feature is skewed like this, some modeling algorithms will make predictions on higher values of the feature

VISUALIZE YOUR DATA: BIVARIATE PLOTTING

- Visualize relationships between 2 data columns. Our GOAL is to understand if 2 columns are correlated with each other.
- GENERAL RULE: don't use highly correlated features in a model.
- Our plotting efforts will center on the bivariate relationship between a single feature and the label of your problem.
  - ↳ will help us explain how our model is working



NUMERIC DATA:

① SCATTER PLOT:

↳ Analyze the CORRELATION between x and y axis.

↳ GENERAL RULE: more dispersion == less correlated

↳ Building & selecting features we want ~0 correlation between individual features.
~1 or ~-1 between features and labels.

↳ NOTE: Don't soley rely on correlational statistic

② BAR PLOT

HACK: By binning the data and plotting, we can compute label rates, get the error bars, and visualize the underlying trend.

BIVARIANT PLOTS: WHEN ONE OF THE VARIABLES IS A LABEL

EX)

Label average for each value of the categorical feature →

⇓

HIGHLY PREDICTIVE FEATURE

- FEATURE is CATEGORICAL
- LABEL is a BINARY OUTCOME
- This is a BARPLOT
- Seaborn automatically include ERROR BARS to show that the differences are statistically significant.
- Plots like this helps build insight around a particular problem.

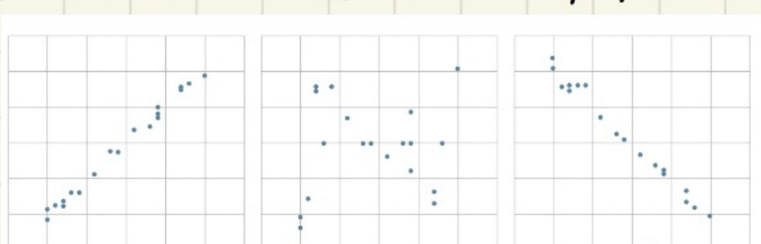
CORRELATION, COVARIANCE, & MUTUAL INFORMATION

Both covariance and correlation find the linear dependencies between variables.

As part of the EDA we're interested in the dependencies between various random variables. Which is commonly acheived by calculating the correlation & mutual information between these variables.

CORRELATION & COVARIANCE:

- For 2 randomly distributed variables the covariance formula is: $Cov(x,y) = \frac{\sum_{i=0}^{N}(x_i-\overline{x})(y_i-\overline{y})}{N-1}$ — Tells us whether two features are directly dependent or inversely dependent on each other.
  - ↳ Larger magnitude of covariance == variables are highly independant on each other
  - ↳ covariance closer to 0 == variables are less linearly dependent on each other.



↑ HIGH POSITIVE COVARIANCE ∴ HIGH LINEAR DEPENDENCY

↑ NEAR 0 COVARIANCE ∴ LITTLE LINEAR DEPENDENCY

↑ HIGH NEGATIVE COVARIANCE ∴ HIGH LINEAR DEPENDENCY

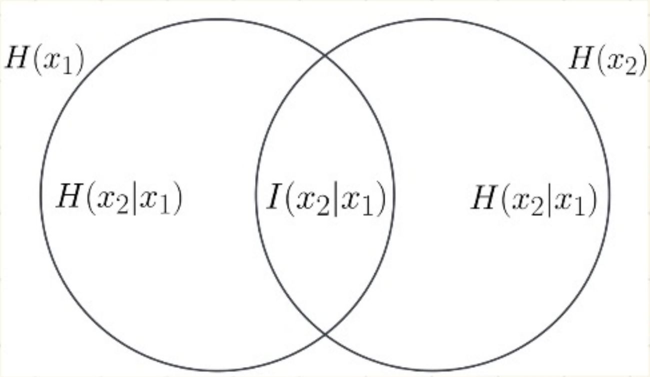
- Magnitude is unbounded & can be arbitrarily large depending on the data you're working on.
- PEARSON CORRELATION: standardizes the range of value for covariance to be always -1 and 1.

- Widely used correlation formula: $Corr(x,y) = \frac{\sum_{i=0}^{N}(x_i - \overline{x})(y_i - \overline{y})}{\sigma_x \sigma_y}$
- By standardizing the covariance we can easily compare the degree of linear dependence between variables.

MUTUAL INFORMATION: Helps us understand the amount of reduction in uncertainty that one random variable provides for another.

- Takes on the range between 0 & 1 & is built upon the concept of uncertainty.
- CONCEPT OF UNCERTAINTY: quantified in information theory as ENTROPY (denoted as H)
  - ↳ Entropy = 0 ⟹ variable is completely predictable
  - ↳ Entropy = 1 ⟹ variable is completely uncertain
- Calculate mutual information between 2 variables: $I(x_1, x_2) = H(x_1) - \underbrace{H(x_1|x_2)}_{\text{uncertainty of variables } x_1 \text{ given variable } x_2.} = H(x_2) - H(x_2|x_1)$

EX)
- Assume $H(x_1)$ is completely uncertain (1)
- Then we introduced $H(x_2)$ which helps bring down the entropy down to a value 0.1 denoted by $H(x_1|x_2)$, meaning our mutual information is: $1 - 0.1 = 0.9$.
  - ↳ b/c x_2 greatly reduces the uncertainty of x_1 ∴ there's high mutual information between the 2 variables.



← If we have large overlapping region between $H(x_1)$ and $H(x_2)$ then we have high mutual information.

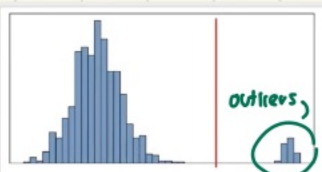
← If variables are completely non-overlapping, then there's no mutual information between 2 variables.

HOW TO USE CORRELATION & MUTUAL INFORMATION

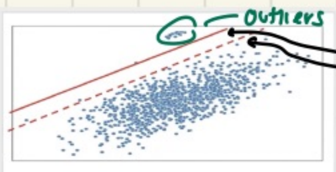
- When we have a large number of features to work with, we only want to choose those that are MOST RELEVANT in predicting our label.
- Too many features ⟹ increases computing cost ⟹ reduces generalizability of our model as it would try to learn from features that are less relevant. ↳ NOT OUR GOAL!
- PROPER APPROACH: select features w/ highest correlation & mutual information against our label.
  Consider any pairs of features that are highly correlated or have high mutual information to be redundant, in which case we may choose to remove one of the features from our dataset.

OUTLIERS: A data point that's far from all other datapoints.

EX)



UNIVARIANT HISTOGRAM



BIVARIANT SCATTER PLOT

- Outliers can be BOTH univariant & multivariant in nature.
- Outliers are subjective; the 2 lines can either present a cutoff point for considering if something is an outlier.
- OUTLIER THRESHOLDS: selection rule of thumb: only consider no more than the top 1% to be an outlier.

REASONS WHY THEY'RE PROBLEMATIC: outliers tend to skew mean values towards the outlier & the increase the variance which makes error bars larger.

↳ Any extrapolation around these errors are likely to be error prone.

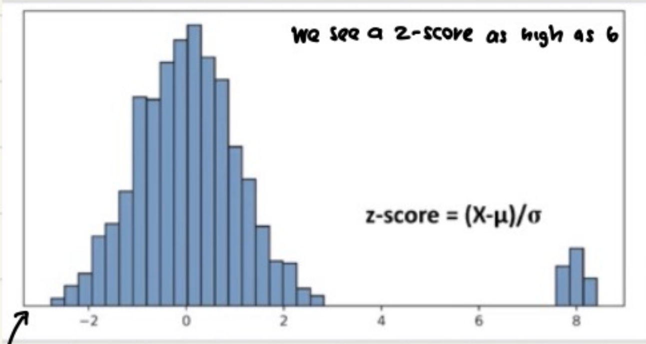
- Outlier can be a clue that something went wrong in the data collection process.
- Researching the outlier can expose some errors in the overall data processing.

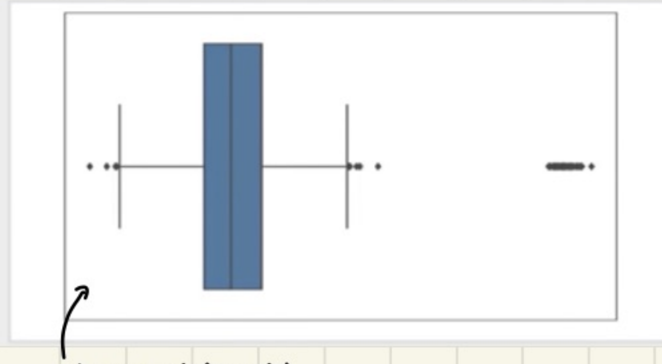
METHODS IN DETECTING OUTLIERS:

Both use the natural variance of a data & outliers are defined based on how much they deviate from normal variance.

① Z-Score: computes z-score for each point and any point with abs(Z) > K is an outlier.
  - ↳ z-score is the point minus the average for that feature divided by the standard deviation.
  - ↳ The absolute value of the z-score is greater than some threshold K ⟹ outlier

② Interquartile range: computes IQR as distance between 25th & 75th percentile. Any point that's K times greater than IQR from 25th or 75th percentile is considered an outlier.





- z-score on the x axis
- z-score is the result of transforming a variable by subtracting the mean from each point & dividing by the standard deviation.
- In a normally distributed variable, most of the mass should have a z-score less than 3.

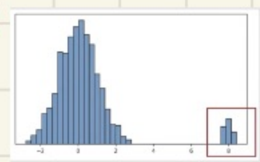
- Box and whiskers plot
- The whiskers are tuned to 2x the interquartile range which is between 25th and 70th percentile.

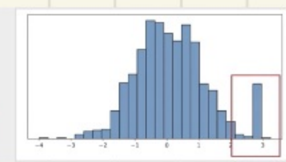
We would want a outlier threshold to determine what's an outlier & what isn't.

WHAT TO DO WITH AN OUTLIER?

1. Discard those examples [if outlier is rare & you suspect it's driven by an error w/ the data generating process]
2. Winsorization:
   - identify the outliers
   - Replace them with a high but acceptable values

Ex)





COMMON STATISTICS REFRESHER

Mean

A mean is an average of a series of numerical values obtained by adding up the numbers and diving by the number of values. There are other kinds of means (geometric, harmonic) that are calculated differently, but the term "mean" usually refers to the arithmetic mean, which is the sum of all the observations divided by the number of observations.

Median

A median is the middle number in a series of numerical values sorted from lowest to highest. If there is an even number of values in the series, the median is the mean (halfway point) of the two middle numbers.

Mode

A mode is the most frequent value in a series. If a series has more than one mode, it's considered multimodal. Multimodality is best demonstrated with a histogram that has two peaks, where each peak can be considered a "local" mode.

Outlier

An outlier is a data point that differs significantly from other observations. This could be due to error (like putting a decimal in the wrong spot) or a true pattern that needs further investigation. There are different statistical ways of identifying outliers and a common one is to flag values that are less than or greater than $1.5 \times IQR$.

Z-Score

The z-score is a way of translating data to understand how many standard deviations away from the mean each point is. If a data point has a z-score of 1, that means it is 1 standard deviation above the mean. If a data point has a z-score of -1.5, that means it is 1.5 standard deviations below the mean. The z-score for each point is calculated by subtracting a value by the mean and dividing by the standard deviation. This is shown in the equation $Z = \frac{x-\mu}{\sigma}$: where x is the value, μ is the mean, and σ is the standard deviation.

Winsorization

Outliers can be removed from a dataset (and often are), or they can be modified in some way to prevent them from having a disproportional impact on the characteristics you are trying to describe in your data. Winsorization is the method of clamping outlier data points at specific values derived from the data itself, like a percentile. Data can be Winsorized from both tails of its distribution by choosing a lower/upper percentile and capping points at those percentiles.

Univariate

Data is univariate if it contains one variable (uni = 1, variate = variable). An example of univariate data is measured dog heights without any other demographic information (e.g., age, breed).

Multivariate

Data is multivariate if it contains more than one variable. An example of multivariate data is measured dog heights along with age, sex, weight, and breed.

Outlier

An outlier is a data point that differs significantly from other observations. This could be due to error (like putting a decimal in the wrong spot) or a true pattern that needs further investigation. There are different statistical ways of identifying outliers and a common one is to flag values that are less than or greater than $1.5 \times IQR$.

Z-Score

The z-score is a way of translating data to understand how many standard deviations away from the mean each point is. If a data point has a z-score of 1, that means it is 1 standard deviation above the mean. If a data point has a z-score of -1.5, that means it is 1.5 standard deviations below the mean. The z-score for each point is calculated by subtracting a value by the mean and dividing by the standard deviation. This is shown in the equation $Z = \frac{x-\mu}{\sigma}$: where x is the value, μ is the mean, and σ is the standard deviation.

Winsorization

Outliers can be removed from a dataset (and often are), or they can be modified in some way to prevent them from having a disproportional impact on the characteristics you are trying to describe in your data. Winsorization is the method of clamping outlier data points at specific values derived from the data itself, like a percentile. Data can be Winsorized from both tails of its distribution by choosing a lower/upper percentile and capping points at those percentiles.

Univariate

Data is univariate if it contains one variable (uni = 1, variate = variable). An example of univariate data is measured dog heights without any other demographic information (e.g., age, breed).

Multivariate

Data is multivariate if it contains more than one variable. An example of multivariate data is measured dog heights along with age, sex, weight, and breed.

OUTLIER DETECTION METHOD

Possible reasons why outliers occur:
- Human error (Ex. entry error)
- System error (Ex. integer overflow)
- Legitimate value, but is unrepresentative of the typical scenario.

2 common approaches for detecting & separating out outliers:

① **Z-score:** The idea of z-score is to take a data point x, subtract it by the mean of the dataset, and then divide it by the standard deviation of the dataset. Typically when a data point has an absolute value of the z-score above 3, it is considered an outlier. In a normal distribution, 99.7% of all points in a dataset are expected to fall within a z-score of 3. Z=3 is also an acceptable threshold, but the MLE can use any threshold and test them.

$$z = \frac{x - \bar{x}}{\sigma}$$

② **IQR:** IQR or Interquartile Range seeks to identify a range where the majority of the data points lie.

IQR is defined by the following formula:

median of the upper half of the data set

IQR = Q3 - Q1

median of the lower half of the dataset.

↳ After finding IQR we then procede to calculate the range of acceptable values by taking:
- ↳ Q1+K*IQR to get the LOWEST ACCECPTABLE VALUE
- ↳ Q3+K*IQR to get the HIGHEST ACCECPTEABLE VALUE

EX) We have an array: [1, 10, 35, 37, 38, 40, 45, 46, 50, 84, 85]. The median = 40.
- ↳ Lower half: [1, 10, 35, 37, 38]; ½ is: 35 = Q1
- ↳ Upper half: [45, 46, 50, 84, 85]; ½ is: 50 = Q3

∴ IQR = 50-35 = 15
K=2

∴ K*IQR = 30 ∴ 30 - Q1 = 5
30 + Q3 = 80

5 & 80 are outliers

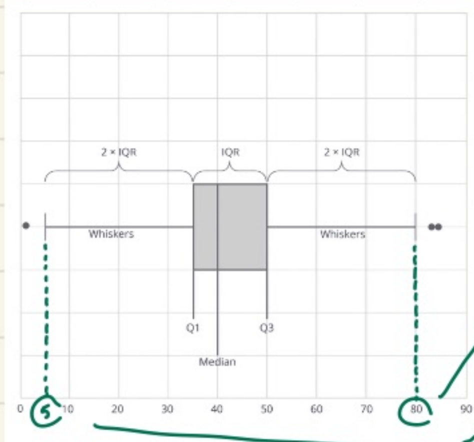


Figure 2. Box and Whiskers Plot

MISSING DATA

If there's consistent errors in your data you probably want to trace the source and fix it.

- Common causes for missing data:
  - → system failures
  - → timeouts from logging data
  - → illegal values being parsed into formatted data field
  - → incomplete webforms/surveys, etc.

HOW TO WORK WITH MISSING DATA:

- Traditional databases use: null
- Pandas data looks like: NaN (not a number) datatype
- There's a possibility that the missing values are acceptable condition by the system and some developer encoded them in a special way.

To examine if a particular column has missing values (T/F): data_frame.isna().sum() >0

STEP 1: Check for missing-ness.
- ↳ Which column has missing values?
- ↳ How many missing values are there?

Must address them!

HANDLING MISSING VALUES:

LOWEST EFFORT: Drop the record or column [DELETION]
- ↳ If missing value count is very low. OR
- ↳ If there's missing values in the same example ← indicates maybe systematic error in the data collection

MODEST EFFORT: Replace missing value with mean or median [IMPUTATION]
- ↳ Often choose MEDIAN when data is highly skewed.

MOST EFFORT: Predict missing value with E[x|x'] [INTERPOLATION]
- ↳ When there's correlation amongst your features, you might be able to predict the real value from the other features.
- ↳ We treat the future of interest as a LABEL
- ↳ Treat other features as PREDICTOR FEATURES

SUBJECTIVITY WHEN HANDLING MISSING VALUES:

- Making choices on the method & the values we might use for imputation if we use imputation.