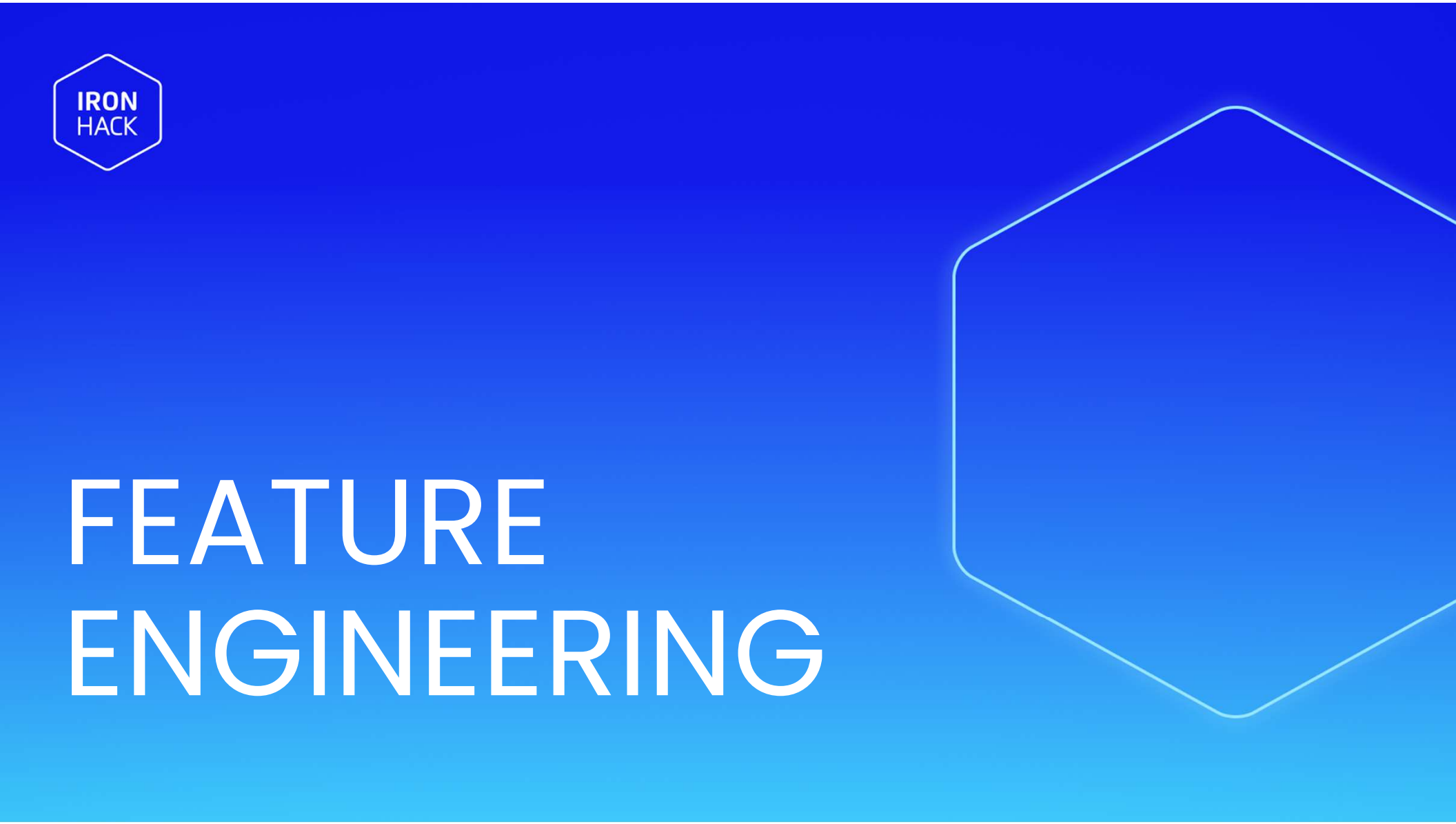

IRON
HACK
FEATURE
ENGINEERING

FEATURE SELECTION – NUMERICAL VS CATEGORICAL

- The techniques used to handle numerical and categorical variables in preparing a dataset for machine learning, are quite different
- Numerical Features are usually used by the algorithms to quantify "distances" between different data points
- On the other hand, we cannot define such distances for categorical variables

If a feature is the colour of an object, how "different" **red** from **blue**?
how about **red** from **green**?

FEATURE SELECTION – ONE HOT ENCODING

- One hot encoding: replace a categorical variable with one of more new features that have the values of 0 or 1.
- What are the consequences of this technique in terms of the cardinality of the variable?

	age	workclass	education	gender	hours-per-week	occupation	income
0	39	State-gov	Bachelors	Male	40	Adm-clerical	<=50K
1	50	Self-emp-not-inc	Bachelors	Male	13	Exec-managerial	<=50K
2	38	Private	HS-grad	Male	40	Handlers-cleaners	<=50K
3	53	Private	11th	Male	40	Handlers-cleaners	<=50K
4	28	Private	Bachelors	Female	40	Prof-specialty	<=50K
5	37	Private	Masters	Female	40	Exec-managerial	<=50K
6	49	Private	9th	Female	16	Other-service	<=50K
7	52	Self-emp-not-inc	HS-grad	Male	45	Exec-managerial	>50K
8	31	Private	Masters	Female	50	Prof-specialty	>50K
9	42	Private	Bachelors	Male	40	Exec-managerial	>50K
10	37	Private	Some-college	Male	80	Exec-managerial	>50K

FEATURE SELECTION – ONE HOT ENCODING

- One hot encoding for the working class column

workclass	Government Employee	Private Employee	Self Employed	Self Employed Incorporated
Government Employee	1	0	0	0
Private Employee	0	1	0	0
Self Employed	0	0	1	0
Self Employed Incorporated	0	0	0	1

- Watch out that numbers can encode categorical variables

FEATURE SELECTION – ONE HOT ENCODING

- Watch out that numbers can encode categorical variables

	Categorical Feature	Integer Feature
0	socks	0
1	fox	1
2	socks	2
3	box	1

	Integer Feature_0	Integer Feature_1	Integer Feature_2	Categorical Feature_box	Categorical Feature_fox	Categorical Feature_socks
0	1.0	0.0	0.0	0.0	0.0	1.0
1	0.0	1.0	0.0	0.0	1.0	0.0
2	0.0	0.0	1.0	0.0	0.0	1.0
3	0.0	1.0	0.0	1.0	0.0	0.0

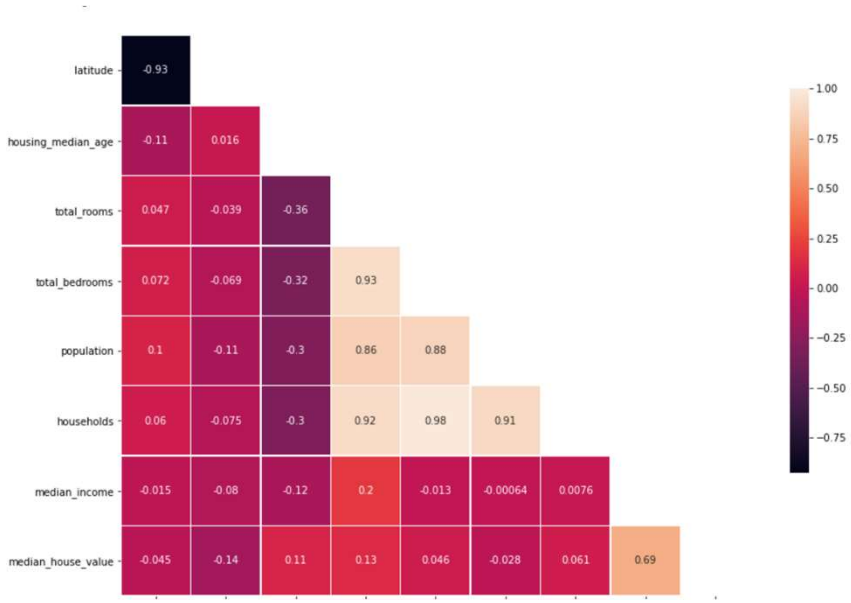
FEATURE SELECTION - BINNING

- Create groups of lower granularity from elements of a variable
- Also known as descritization when applied to numerical variables
- In Python can be done with cut and qcut

FEATURE SELECTION – TESTS OF INDEPENDENCE

- Remember Module 2 techniques:
- - Chi-Squared goodness of fit tests of contingency
- - 2-sample t-tests (and ANOVA tests) to see if you can perform further aggregations

CORRELATION THRESHOLDS

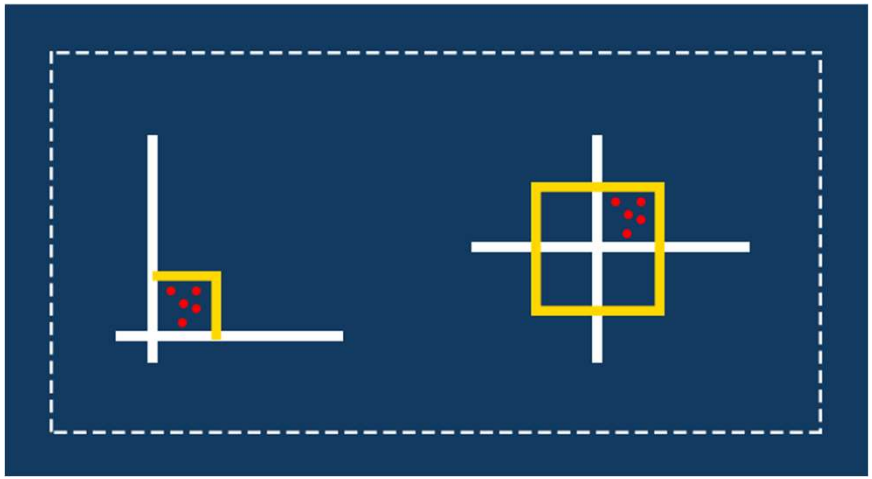


Deep relation with the concept of mutual information

The best numerical features are those with a high explanatory power of the target variable, but low mutual information amongst themselves

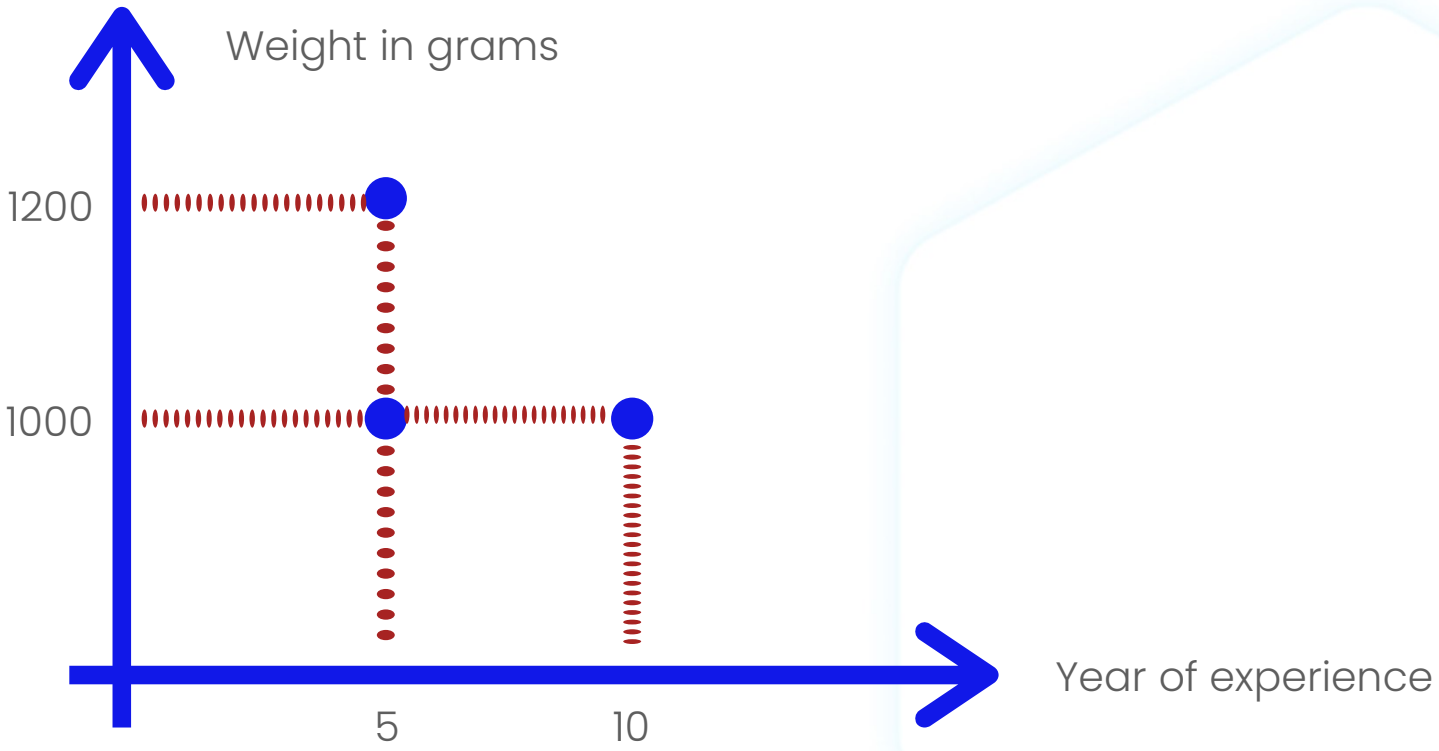
Correlation thresholds allow us create the above criteria and thus filter out redundant variables

Feature Scaling

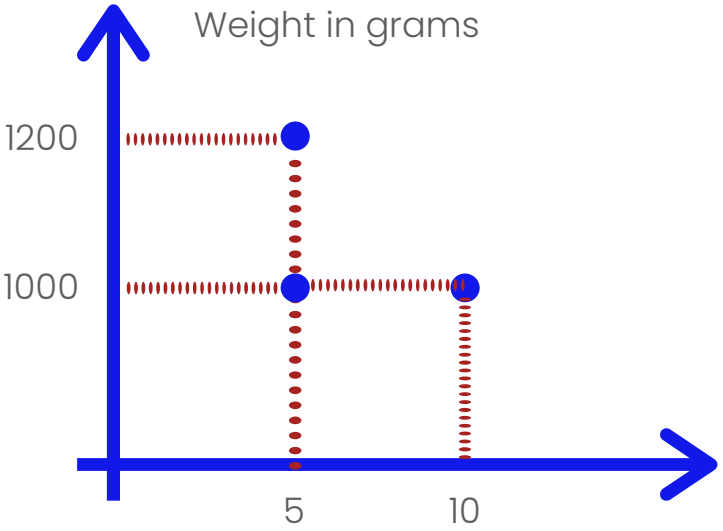


The process of processing numerical features and converting them into a uniform scale is known as feature scaling

whilst some algorithms are virtually invariant to this technique, for other it is determinent

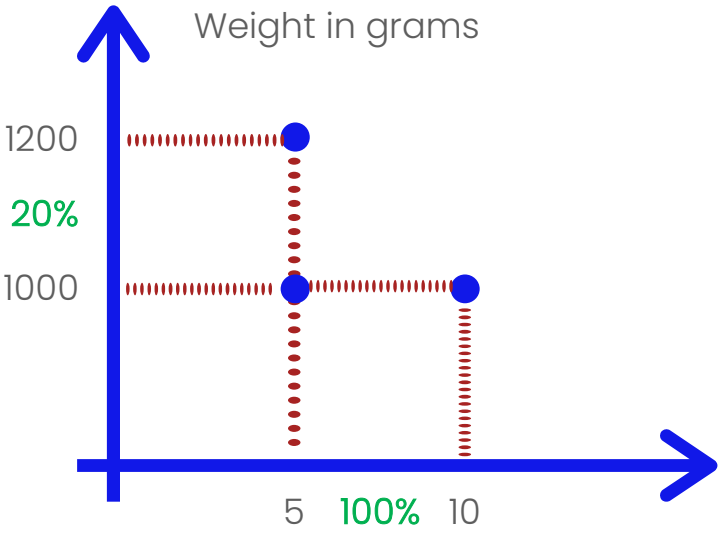

Weight in grams
1200
1000
5
10
Year of experience

FEATURE SCALING – WHY IS IT IMPORTANT?



Which point is closer from a KNN perspective?

FEATURE SCALING – WHY IS IT IMPORTANT?



Crucial in distance based algorithms

NORMALIZATION

Each value of the dataset x will be converted to a new "normalized" value z

$$z = \frac{x \text{ - } \min(x)}{\max(x) \text{ - } \min(x)}$$

Shape of the distribution of data is unchanged

NORMALIZATION

RE-SCALE THE DISTRIBUTION TO BE BETWEEN ZERO AND ONE!

EACH VALUE OF THE DATASET X WILL BE CONVERTED TO A NEW "NORMALIZED" VALUE Z

$$z = \frac{x \text{ - } \min(x)}{\max(x) \text{ - } \min(x)}$$

SHAPE OF THE DISTRIBUTION OF DATA IS UNCHANGED

Z-SCORE : STANDARDIZATION

The z-score is a way to standardize/normalize all your data in a way that tells you how many standard deviations each point is from the mean

If the z-score is **less than 1:** that data point is **within 1 standard deviation** of mean

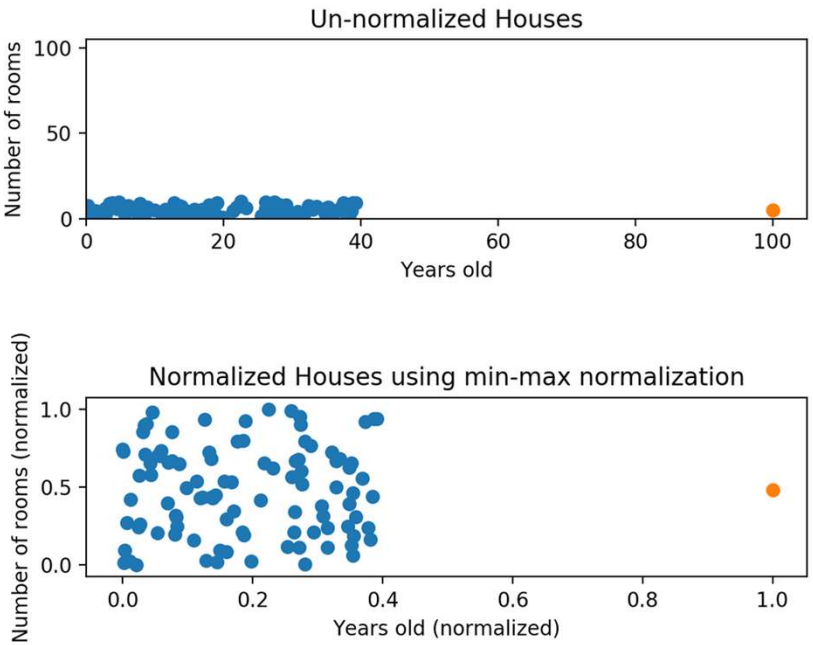
Allows you to bring different distributions to the same scale

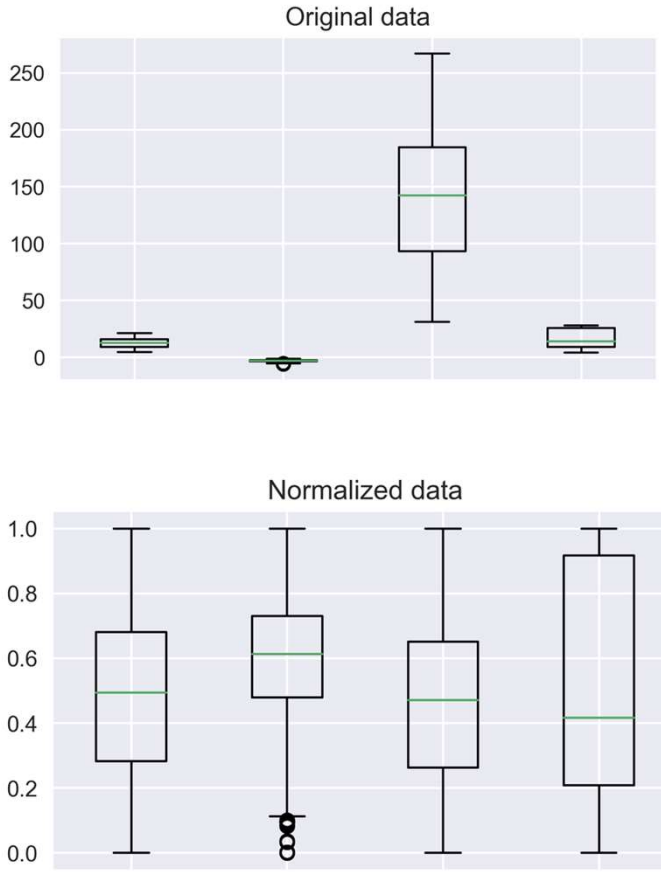
$$Z = \frac{x - \mu}{\sigma}$$

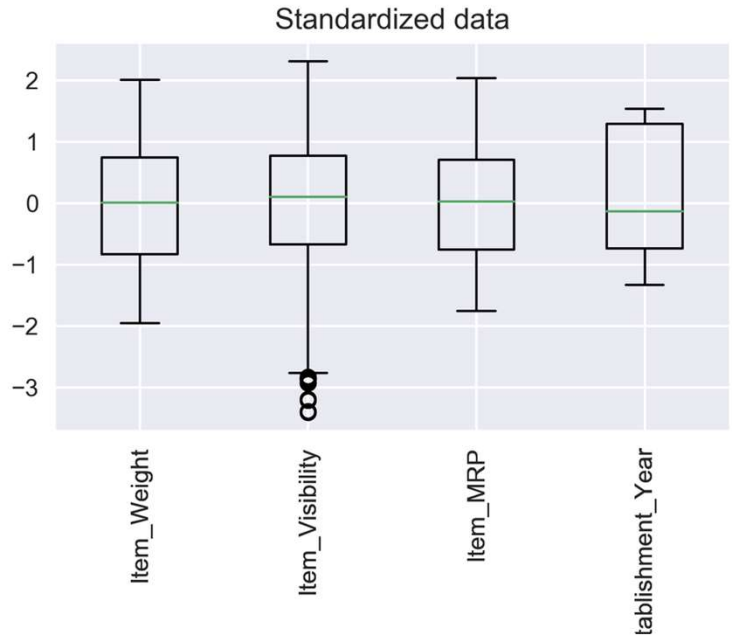
Example of usage: to standardize grades across different schools

Very important in machine learning

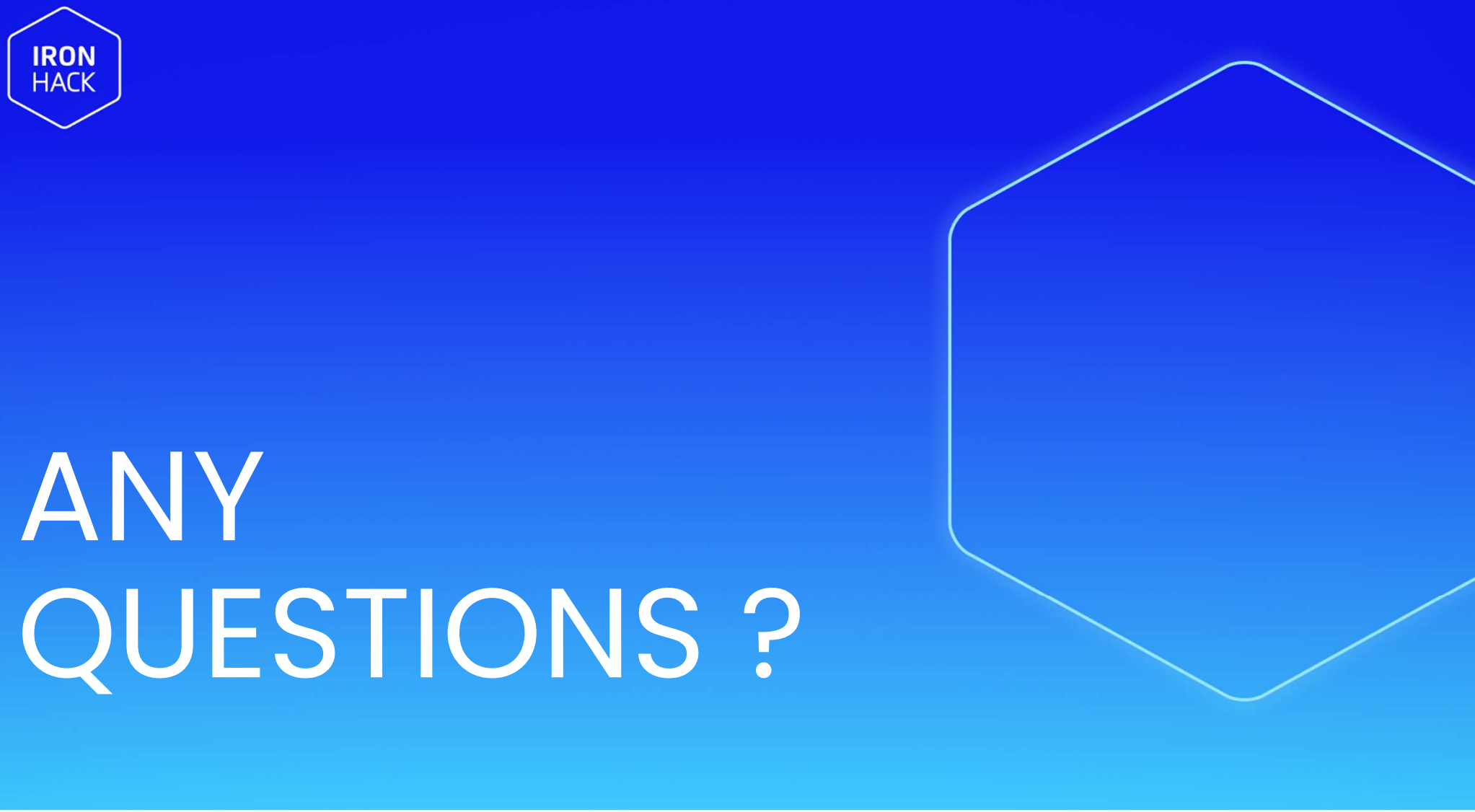
BEFORE AND AFTER FEATURE SCALING



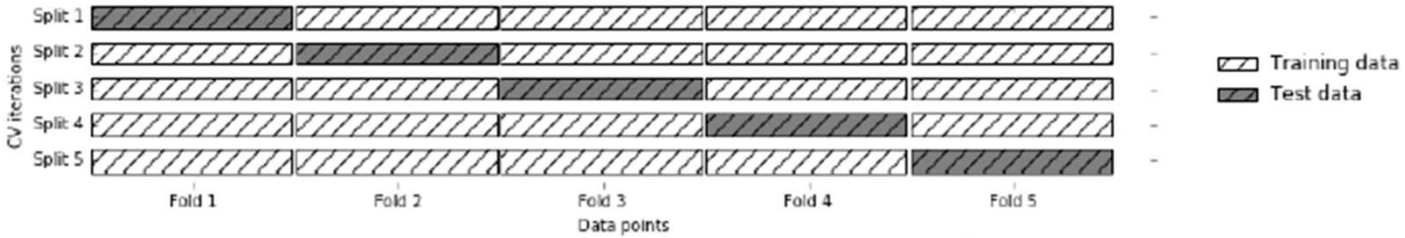

Original data
250
200
150
100
50
0
Normalized data
1.0
0.8
0.6
0.4
0.2
0.0


Standardized data
2
1
0
−1
−2
−3
Item_Weight
Item_Visibility
Item_MRP
tablishment_Year

Feature Engineering – Increase Signal


IRON
HACK
ANY
QUESTIONS ?

MODEL GENERALIZATION



The main disadvantage of cross-validation is increased computational cost. As we are now training k models instead of a single model, cross-validation will be roughly times slower than doing a single split of the data.

It is important to keep in mind that cross-validation is not a way to build a model that can be applied to new data. Cross-validation does not return a model. When calling **cross_val_score**, multiple models are built internally, but the purpose of cross-validation is only to evaluate how well a given algorithm will generalize when trained on a specific dataset.

GENERALIZATION, UNDERFITTING & OVERFITTING