

Feature Engineering for ML

Dmitry Larko
Sr Data Scientist
H2O.ai

[@DmitryLarko](https://twitter.com/DmitryLarko)



About me



Dmitry Larko

Sr. Data Scientist at H2O.ai
San Francisco Bay Area, CA, United States
Joined 6 years ago · last seen in the past day

  <https://h2o.ai>

Followers 163
Following 1



Competitions
Grandmaster

[Home](#) [Competitions \(41\)](#) [Kernels \(0\)](#) [Discussion \(32\)](#) [Datasets \(0\)](#) ...

[Edit Profile](#)

Competitions Grandmaster



Current Rank
78
of 82,719

Highest Rank
25


10


12


7

Kernels Novice



Unranked


0


0


0

Discussion Novice



Unranked


0


3


13



Feature Engineering

"Coming up with features is difficult, time-consuming, requires expert knowledge. "Applied machine learning" is basically feature engineering." ~ Andrew Ng

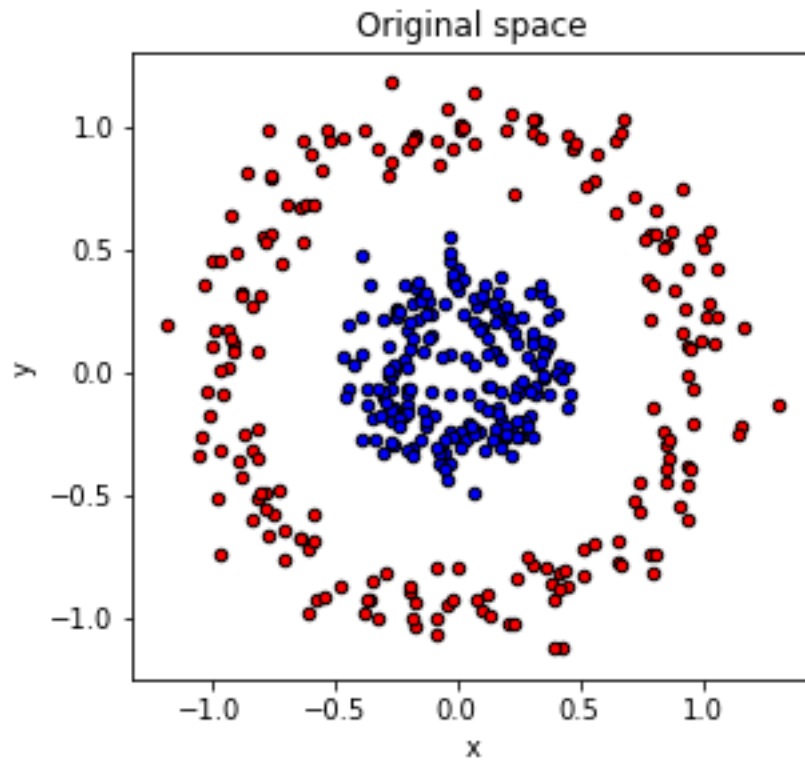
"... some machine learning projects succeed and some fail. What makes the difference? Easily the most important factor is the features used." ~ Pedro Domingos

"Good data preparation and feature engineering is integral to better prediction" ~ Marios Michailidis (KazAnova), Kaggle GrandMaster, Kaggle #3, former #1

"*you have to turn your inputs into things the algorithm can understand*" ~ Shayne Miel, answer to ["What is the intuitive explanation of feature engineering in machine learning?"](#)

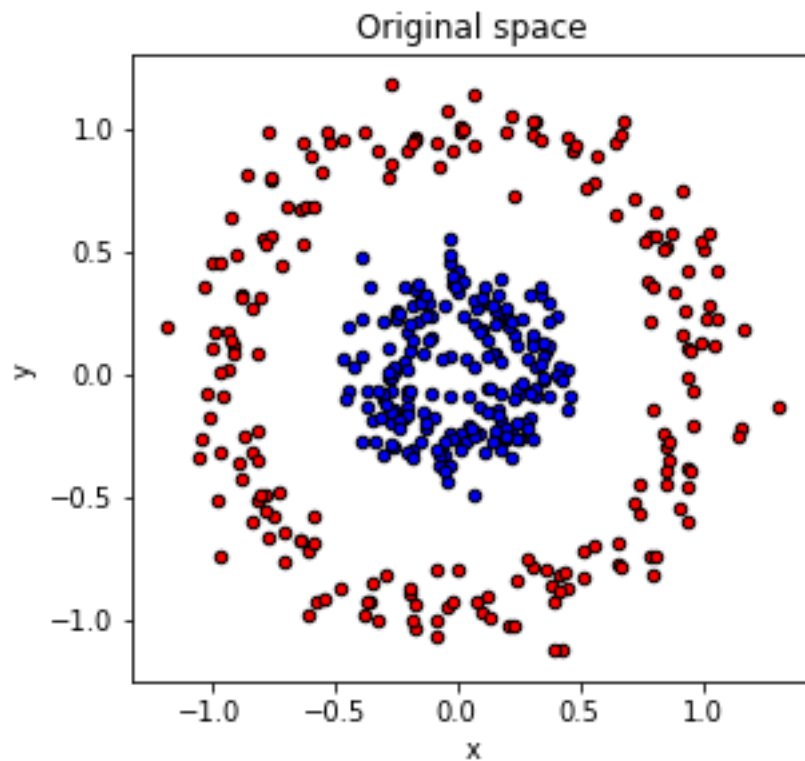


What is feature engineering



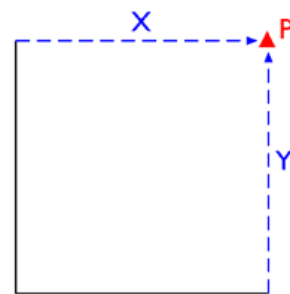
Not possible to separate using linear classifier

What is feature engineering



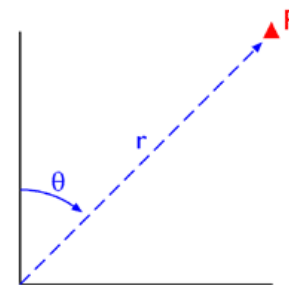
What if we use polar coordinates instead?

Cartesian coordinates



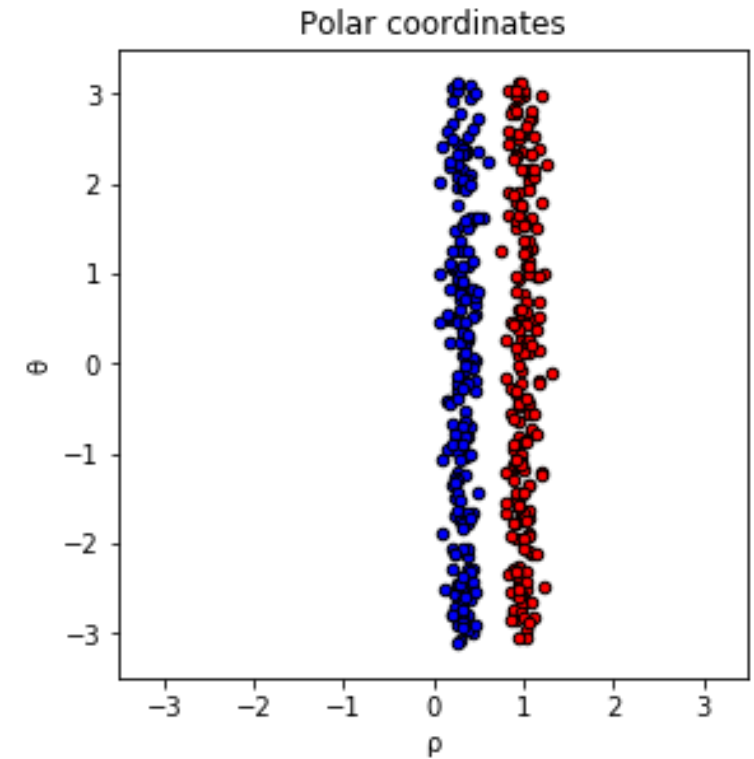
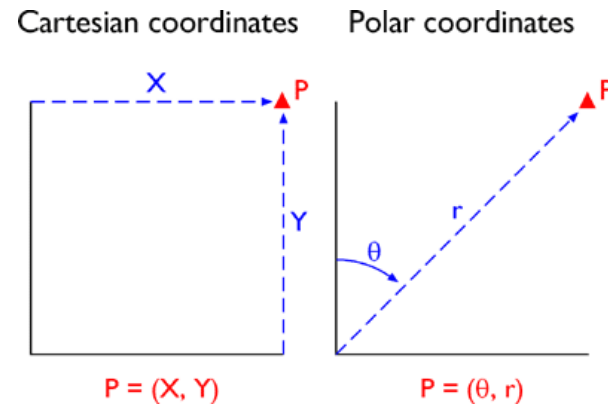
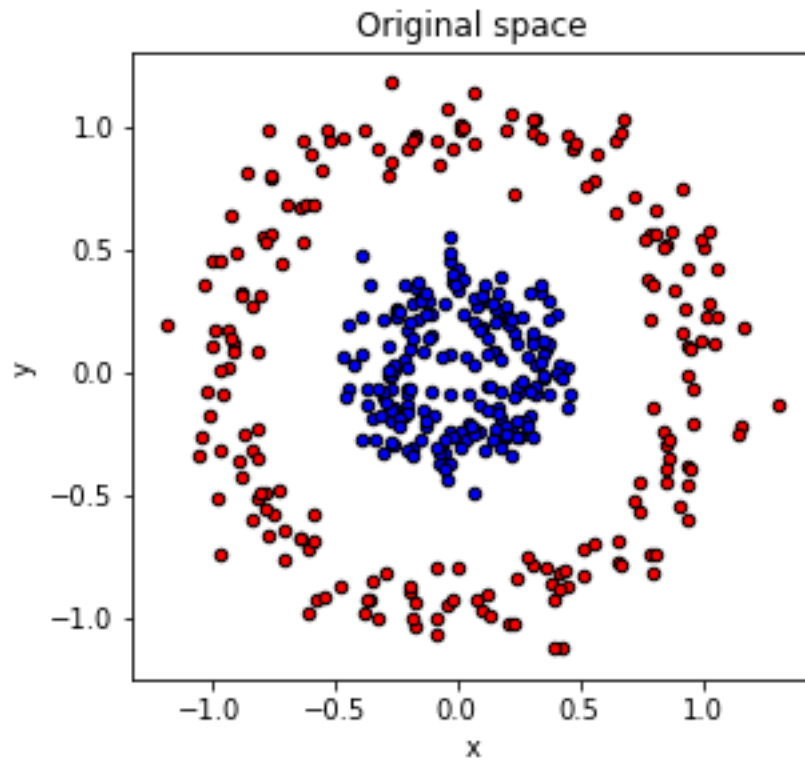
$P = (X, Y)$

Polar coordinates

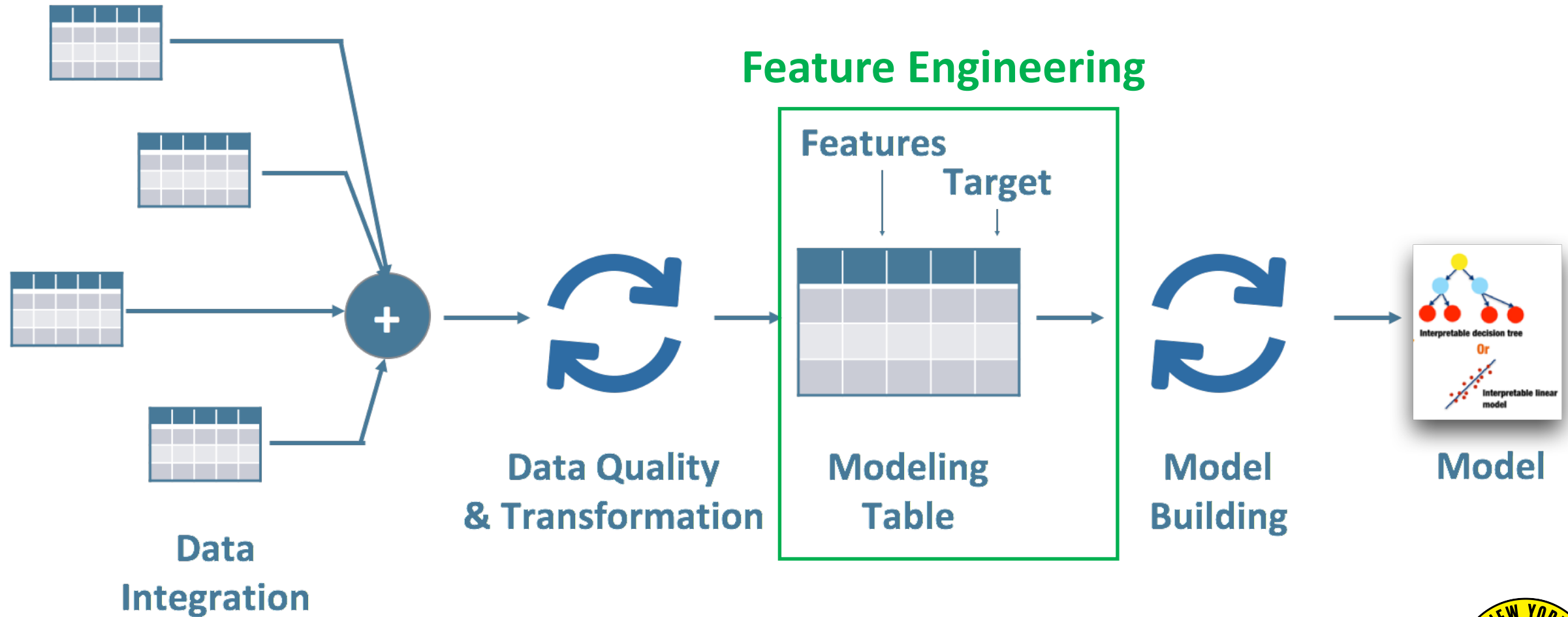


$P = (\theta, r)$

What is feature engineering



Typical Enterprise Machine Learning Workflow



Your data

Return%	ProductID	Dept	Price	MFR
1.94	54323	Household	54.95	USA
0.023	92356	Household	9.95	USA
0.8	78023	Computer	4.5	China
0.01	12340	Audio	109.99	China
0.41	31240	Audio	29.99	Taiwan
0.97	12351	Hardware	54.95	Mexico
0.0115	90141	Hardware	4.99	USA
0.4	81240	Hardware	6.55	Taiwan
0.03	14896	Computer	211.99	Korea
0.205	62132	Computer	1100	USA
1.6878	54323	Audio	34.99	USA
0.0345	92356	Audio	7.99	USA
0.64	78023	Household	229.9	Brazil
0.72	12340	Audio	19.95	Mexico
0.41	31240	Computer	6.99	Taiwan
1.94	54323	Hardware	11.99	Taiwan
0.023	92356	Household	2.05	USA
0.08	78023	Computer	99.99	USA
2.09	12340	Computer	129.99	China
1.1	31240	Audio	18.99	China



Your data

Target	Return%	ProductID	Dept	Price	MFR
	1.94	54323	Household	54.95	USA
	0.023	92356	Household	9.95	USA
	0.8	78023	Computer	4.5	China
	0.01	12340	Audio	109.99	China
	0.41	31240	Audio	29.99	Taiwan
	0.97	12351	Hardware	54.95	Mexico
	0.0115	90141	Hardware	4.99	USA
	0.4	81240	Hardware	6.55	Taiwan
	0.03	14896	Computer	211.99	Korea
	0.205	62132	Computer	1100	USA
	1.6878	54323	Audio	34.99	USA
	0.0345	92356	Audio	7.99	USA
	0.64	78023	Household	229.9	Brazil
	0.72	12340	Audio	19.95	Mexico
	0.41	31240	Computer	6.99	Taiwan
	1.94	54323	Hardware	11.99	Taiwan
	0.023	92356	Household	2.05	USA
	0.08	78023	Computer	99.99	USA
	2.09	12340	Computer	129.99	China
	1.1	31240	Audio	18.99	China



Your data

Target	Categorical			Cat
Return%	ProductID	Dept	Price	MFR
1.94	54323	Household	54.95	USA
0.023	92356	Household	9.95	USA
0.8	78023	Computer	4.5	China
0.01	12340	Audio	109.99	China
0.41	31240	Audio	29.99	Taiwan
0.97	12351	Hardware	54.95	Mexico
0.0115	90141	Hardware	4.99	USA
0.4	81240	Hardware	6.55	Taiwan
0.03	14896	Computer	211.99	Korea
0.205	62132	Computer	1100	USA
1.6878	54323	Audio	34.99	USA
0.0345	92356	Audio	7.99	USA
0.64	78023	Household	229.9	Brazil
0.72	12340	Audio	19.95	Mexico
0.41	31240	Computer	6.99	Taiwan
1.94	54323	Hardware	11.99	Taiwan
0.023	92356	Household	2.05	USA
0.08	78023	Computer	99.99	USA
2.09	12340	Computer	129.99	China
1.1	31240	Audio	18.99	China



Your data

Target	Categorical		Num	Cat
Return%	ProductID	Dept	Price	MFR
1.94	54323	Household	54.95	USA
0.023	92356	Household	9.95	USA
0.8	78023	Computer	4.5	China
0.01	12340	Audio	109.99	China
0.41	31240	Audio	29.99	Taiwan
0.97	12351	Hardware	54.95	Mexico
0.0115	90141	Hardware	4.99	USA
0.4	81240	Hardware	6.55	Taiwan
0.03	14896	Computer	211.99	Korea
0.205	62132	Computer	1100	USA
1.6878	54323	Audio	34.99	USA
0.0345	92356	Audio	7.99	USA
0.64	78023	Household	229.9	Brazil
0.72	12340	Audio	19.95	Mexico
0.41	31240	Computer	6.99	Taiwan
1.94	54323	Hardware	11.99	Taiwan
0.023	92356	Household	2.05	USA
0.08	78023	Computer	99.99	USA
2.09	12340	Computer	129.99	China
1.1	31240	Audio	18.99	China



Feature Engineering

- Extract more new gold features, remove irrelevant or noisy features
 - Simpler models with better results
- Key Elements
 - Numerical features
 - Categorical features
 - Target Transformation



Numerical features – missing values

- Some machine learning tools cannot accept NAs in the input
- Binary features
 - -1 for negatives, 0 for missing values and 1 for positives
- Numeric features
 - Tree-based methods
 - Encode as a big positive or negative number
 - 999, -99999,
 - $2 \cdot \max(x)$, $2 \cdot \min(x)$
 - Linear, neural nets, etc. methods
 - Encode by splitting into 2 columns:
 - Binary column isNA (0 if not and 1 if yes)
 - In original column replace NAs by mean or median



Numerical features – missing values

Binary	Numerical
1	NaN
0	6
1	5
NaN	8
0	0
0	NaN
1	9
NaN	3
0	5



Binary	Numerical		
	ver 1: 2*max	ver 2: mean and isNA	
		Value	isNA
1	18	5.14	1
-1	6	6	0
1	5	5	0
0	8	8	0
-1	0	0	0
-1	18	5.14	1
1	9	9	0
0	3	3	0
-1	5	5	0

Categorical features – missing values

- Encode as an unique category
 - “Unknown”, “Missing”,
- Use the most frequent category level

Feature 1	Encoded Feature 1	Encoded Feature 1
A	A	A
A	A	A
NaN	MISSING	A
A	A	A
B	B	B
B	B	B
NaN	MISSING	A
C	C	C
C	C	C



Categorical Encoding

- Turn categorical features into numeric features to provide more fine-grained information
 - Help explicitly capture non-linear relationships and interactions between the values of features
 - Most of machine learning tools only accept numbers as their input
 - xgboost, gbm, glmnet, libsvm, liblinear, etc.



Categorical Encoding

- Labeled Encoding
 - Interpret the categories as ordered integers (mostly wrong)
 - Python scikit-learn: LabelEncoder
 - Ok for tree-based methods
- One Hot Encoding
 - Transform categories into individual binary (0 or 1) features
 - Python scikit-learn: DictVectorizer, OneHotEncoder
 - Ok for K-means, Linear, NNs, etc.



Categorical Encoding

- Labeled Encoding

A	0
B	1
C	2

Feature 1	Encoded Feature 1
A	0
A	0
A	0
A	0
B	1
B	1
B	1
C	2
C	2



Categorical Encoding

- One Hot Encoding

A	1	0	0
B	0	1	0
C	0	0	1

Feature	Feature = A	Feature = B	Feature = C
A	1	0	0
A	1	0	0
A	1	0	0
A	1	0	0
B	0	1	0
B	0	1	0
B	0	1	0
C	0	0	1
C	0	0	1



Categorical Encoding

- Frequency Encoding
 - Encoding of categorical levels of feature to values between 0 and 1 based on their relative frequency

A	0.44 (4 out of 9)
B	0.33 (3 out of 9)
C	0.22 (2 out of 9)

Feature	Encoded Feature
A	0.44
A	0.44
A	0.44
A	0.44
B	0.33
B	0.33
B	0.33
C	0.22
C	0.22



Categorcial Encoding - Target mean encoding

- Instead of dummy encoding of categorical variables and increasing the number of features we can encode each level as the mean of the response.

A	0.75 (3 out of 4)
B	0.66 (2 out of 3)
C	1.00 (2 out of 2)

Feature	Outcome	MeanEncode
A	1	0.75
A	0	0.75
A	1	0.75
A	1	0.75
B	1	0.66
B	1	0.66
B	0	0.66
C	1	1.00
C	1	1.00



Categorical Encoding - Target mean encoding

- Also it is better to calculate weighted average of the overall mean of the training set and the mean of the level:

$$\lambda(n) * mean(level) + (1 - \lambda(n)) * mean(dataset)$$

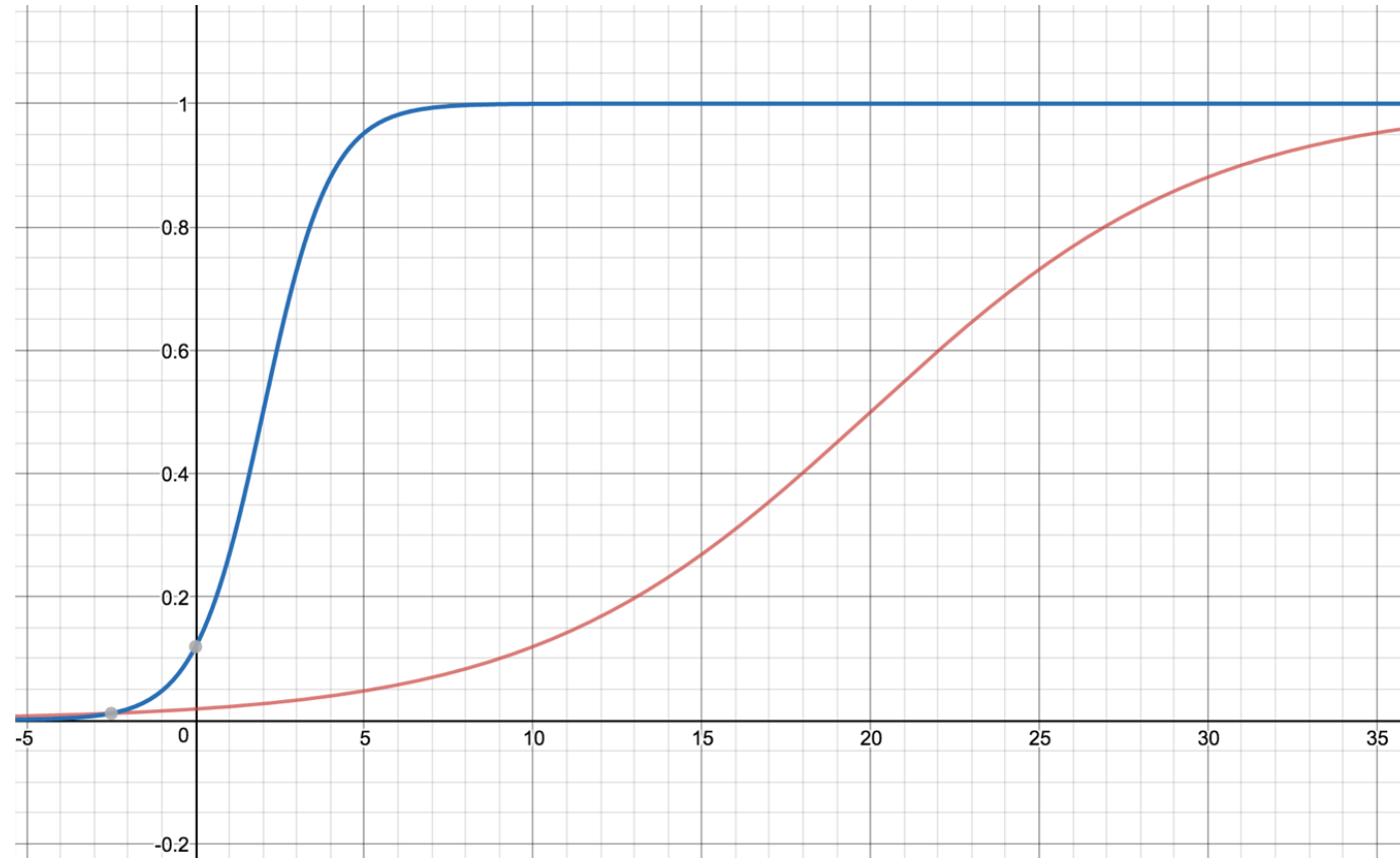
- The weights are based on the frequency of the levels i.e. if a category only appears a few times in the dataset then its encoded value will be close to the overall mean instead of the mean of that level.



Categorical Encoding – Target mean encoding $\lambda(n)$ example

$$\frac{1}{1 + \exp\left(\frac{-(x-k)}{f}\right)}$$

x = frequency
k = inflection
point
f = steepness



Categorical Encoding - Target mean encoding - Smoothing

$$\lambda = \frac{1}{1 + \exp(-\frac{(x - 2)}{0.25})}$$

	x	level	dataset	λ	
A	4	0.75	0.77	0.99	$0.99*0.75 + 0.01*0.77 = 0.7502$
B	3	0.66	0.77	0.98	$0.98*0.66 + 0.02*0.77 = 0.6622$
C	2	1.00	0.77	0.5	$0.5*1.0 + 0.5*0.77 = 0.885$

$$\lambda = \frac{1}{1 + \exp(-\frac{(x - 3)}{0.25})}$$

	x	level	dataset	λ	
A	4	0.75	0.77	0.98	$0.98*0.75 + 0.01*0.77 = 0.7427$
B	3	0.66	0.77	0.5	$0.5*0.66 + 0.5*0.77 = 0.715$
C	2	1.00	0.77	0.017	$0.017*1.0 + 0.983*0.77 = 0.773$

Feature	Outcome
A	1
A	0
A	1
A	1
B	1
B	1
B	0
C	1
C	1



Categorical Encoding - Target mean encoding

- Instead of dummy encoding of categorical variables and increasing the number of features we can encode each level as the mean of the response.

A	0.75 (3 out of 4)
B	0.66 (2 out of 3)
C	1.00 (2 out of 2)

Feature	Outcome	MeanEncode
A	1	0.75
A	0	0.75
A	1	0.75
A	1	0.75
B	1	0.66
B	1	0.66
B	0	0.66
C	1	1.00
C	1	1.00



Categorical Encoding - Target mean encoding

- To avoid overfitting we could use leave-one-out schema

Feature	Outcome		LOOencode
A	1		0.66
A	0		
A	1		
A	1		
B	1		
B	1		
B	0		
C	1		
C	1		



Categorical Encoding - Target mean encoding

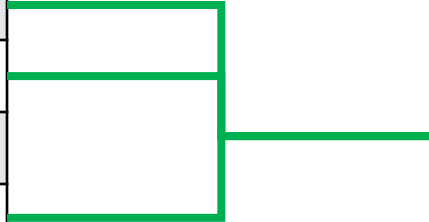
- To avoid overfitting we could use leave-one-out schema

Feature	Outcome	LOOencode
A	1	
A	0	0.66
A	1	1.00
A	1	
B	1	
B	1	
B	0	
C	1	
C	1	



Categorical Encoding - Target mean encoding

- To avoid overfitting we could use leave-one-out schema

Feature	Outcome		LOOencode
A	1		0.66
A	0		1.00
A	1		0.66
A	1		
B	1		
B	1		
B	0		
C	1		
C	1		



Categorical Encoding - Target mean encoding

- To avoid overfitting we could use leave-one-out schema

Feature	Outcome		LOOencode
A	1		0.66
A	0		1.00
A	1		0.66
A	1		0.66
B	1		
B	1		
B	0		
C	1		
C	1		



Categorical Encoding - Target mean encoding

- To avoid overfitting we could use leave-one-out schema

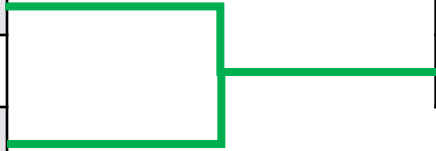
Feature	Outcome	LOOencode
A	1	0.66
A	0	1.00
A	1	0.66
A	1	0.66
B	1	0.50
B	1	
B	0	
C	1	
C	1	



Categorical Encoding - Target mean encoding

- To avoid overfitting we could use leave-one-out schema

Feature	Outcome		LOOencode
A	1		0.66
A	0		1.00
A	1		0.66
A	1		0.66
B	1		0.50
B	1		0.50
B	0		
C	1		
C	1		



LOOencode
0.66
1.00
0.66
0.66
0.50
0.50



Categorical Encoding - Target mean encoding

- To avoid overfitting we could use leave-one-out schema

Feature	Outcome	LOOencode
A	1	0.66
A	0	1.00
A	1	0.66
A	1	0.66
B	1	0.50
B	1	0.50
B	0	1.00
C	1	
C	1	



Categorical Encoding - Target mean encoding

- To avoid overfitting we could use leave-one-out schema

Feature	Outcome
A	1
A	0
A	1
A	1
B	1
B	1
B	0
C	1
C	1

LOOencode
0.66
1.00
0.66
0.66
0.50
0.50
1.00
1.00



Categorical Encoding - Target mean encoding

- To avoid overfitting we could use leave-one-out schema

Feature	Outcome	LOOencode
A	1	0.66
A	0	1.00
A	1	0.66
A	1	0.66
B	1	0.50
B	1	0.50
B	0	1.00
C	1	1.00
C	1	1.00



Categorical Encoding – Numerical features

- Binning using quantiles (population of the same size in each bin) or histograms (bins of same size)
 - Replace with bin's mean or median
 - Treat bin id as a category level and use any categorical encoding schema
- Dimensionality reduction techniques – SVD and PCA
- Clustering and using cluster IDs or/and distances to cluster centers as new features



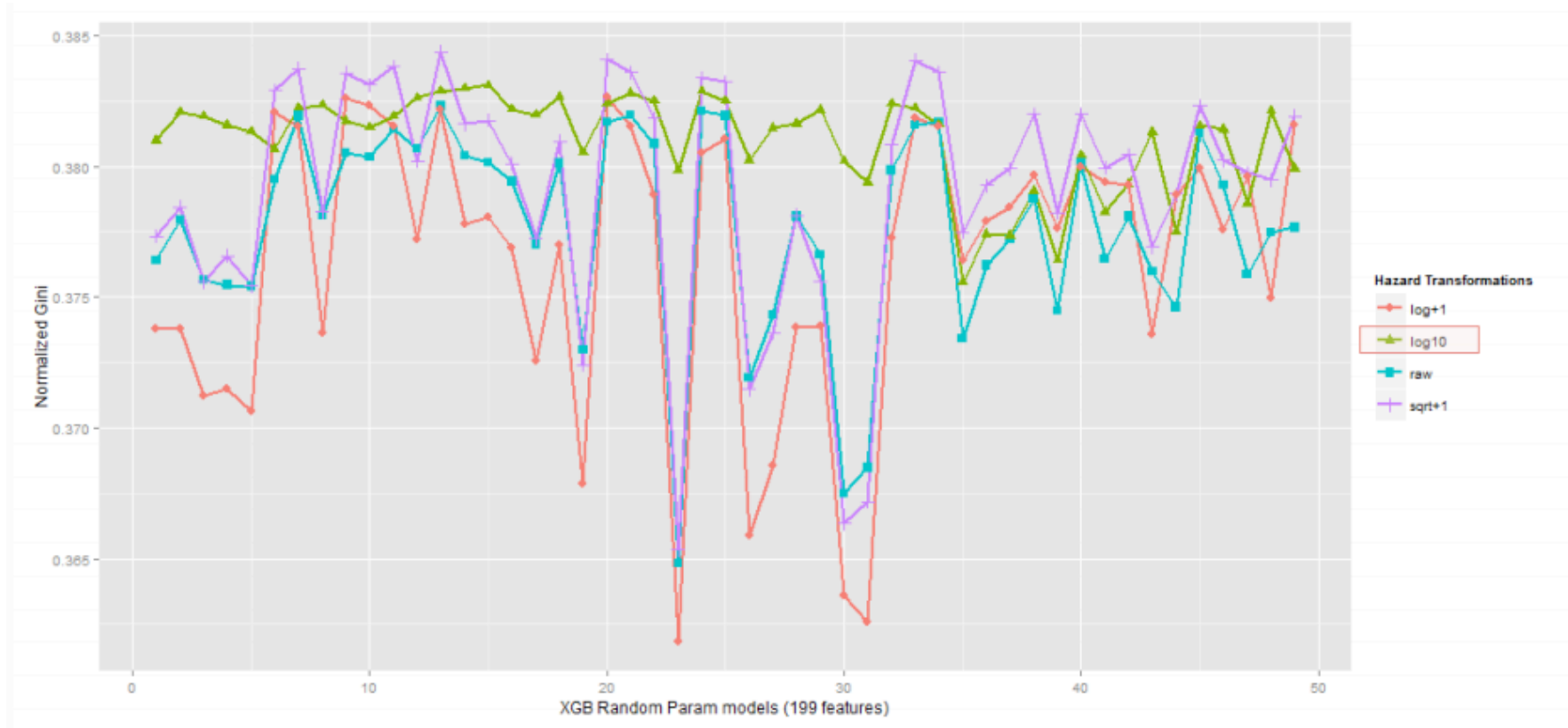
Target Transformation

- Predictor/Response Variable Transformation
 - Use it when variable shows a skewed distribution make the residuals more close to “normal distribution” (bell curve)
 - Can improve model fit
 - $\log(x)$, $\log(x+1)$, \sqrt{x} , $\sqrt{x+1}$, etc.



Target Transformation

In Liberty Mutual Group: Property Inspection Prediction Competition

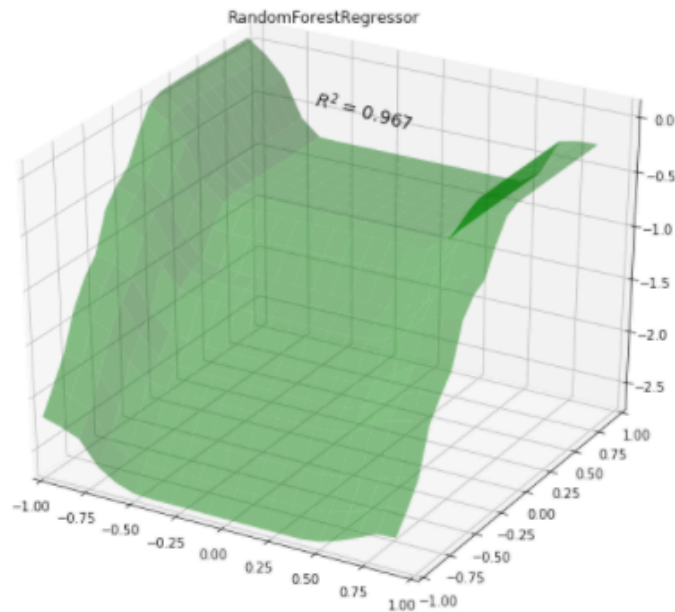
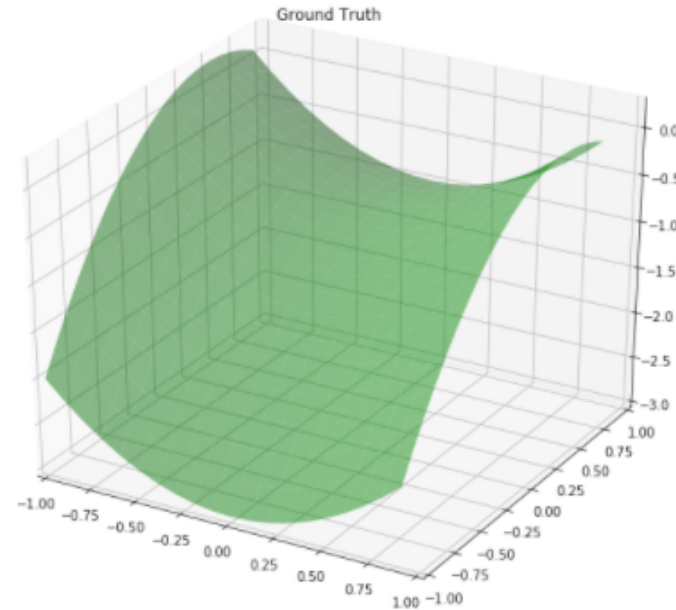


Different transformations might lead to different results

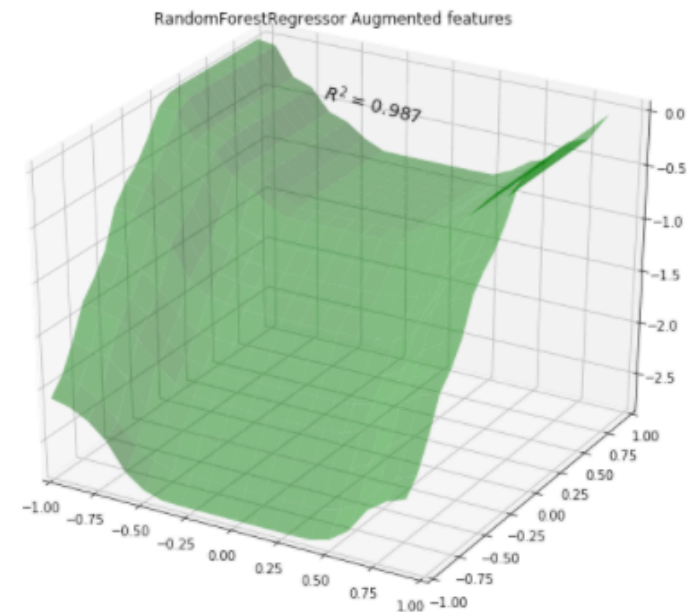
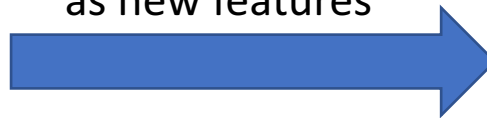


Feature Interaction

- $y = x_1^2 - x_2^2 + x_2 - 1$



Adding x_1^2 and x_2^2
as new features



Feature Interaction – how to find?

- Domain knowledge
- ML algorithm behavior (for example analyzing GBM splits or linear regressor weights)

Feature Interaction – how to model?

- Math operations (sum, div, mul, sub)
- Clustering and kNN for numerical features
- Target encoding for pairs (or even triplets and etc.) of categorical features
- Encode categorical features by stats of numerical features



Thank you!

Q&A

