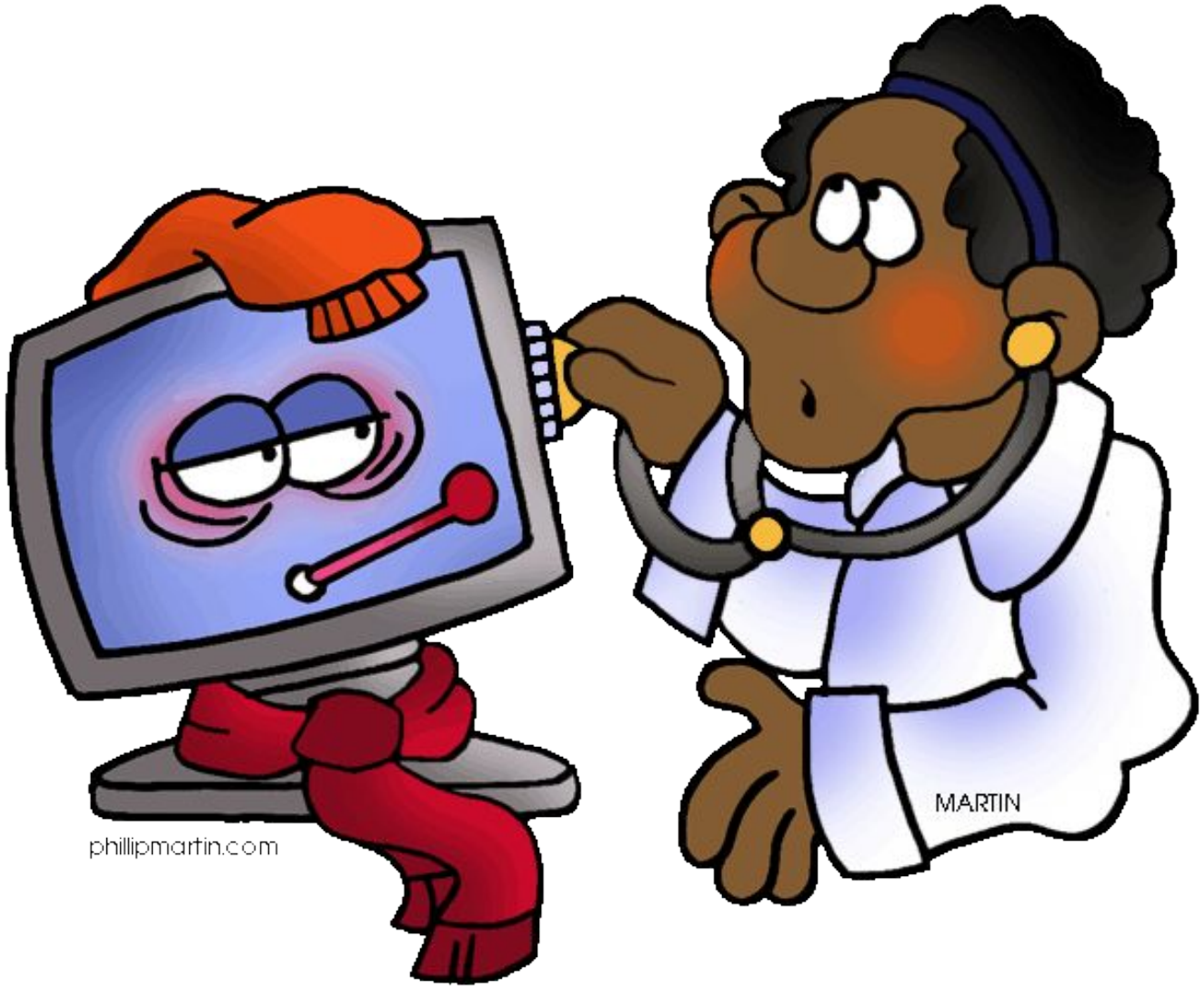
Malware Prediction Challenge

Akshay Punhani | Takuma Kinoshita | Sebastian Gude

Mentor : Henoch Wong

How to figure out if your computer is about to catch a cold?



Can 1 Billion machines be protected from damage?

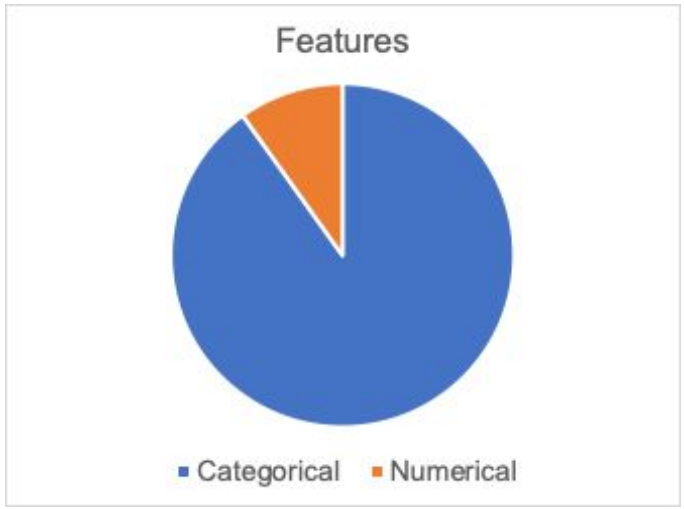
Goal & Motivation

Explore what a typical data science project in the security/fraud detection space looks like	Motivation
Microsoft has more than 1 billion enterprise and consumer customers*	Criticality
Predict a Windows machine's probability of getting infected by various families of malware	End Goal

*https://www.kaggle.com/c/microsoft-malware-prediction

Let’s first take a look at the huge dataset

9 Million Records



81 Total Features

Feature	Total Missing	% Missing
PuaMode	8919174	99.97
Census_ProcessorClass	8884852	99.58
DefaultBrowsersIdentifier	8488045	95.14
Census_IsFlightingInternal	7408759	83.04
Census_InternalBatteryType	6338429	71.04
Census_ThresholdOptIn	5667325	63.52
Census_IsWIMBootEnabled	5659703	63.43

High Percentage of Missing Values

One-hot Encoding vs Label Encoding

One Hot Encoding

Apple	Chicken	Broccoli	Calories
1	0	0	95
0	1	0	231
0	0	1	50

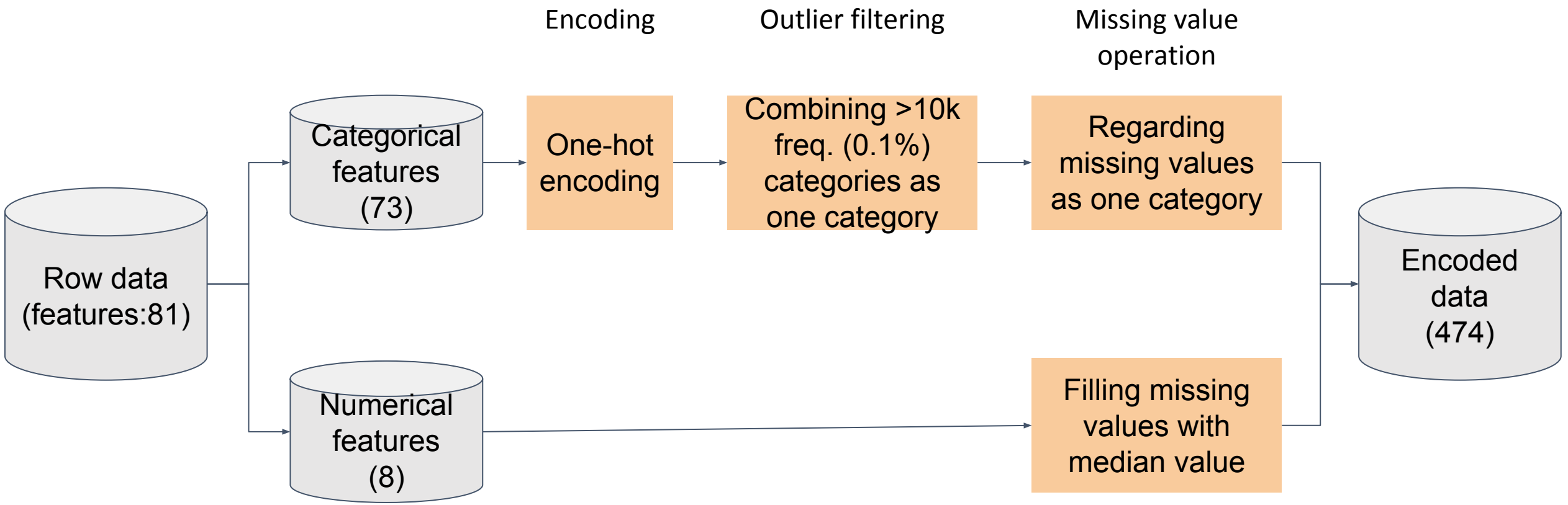
Label Encoding

Food Name	Categorical #	Calories
Apple	1	95
Chicken	2	231
Broccoli	3	50

	One-hot Encoding	Label Encoding
Pros	• More interpretability of model parameters (category level) • Not add meaningless ordinal relationship	• Low memory space • Keep ordinal relationship
Cons	• More memory space • Lose ordinal relationship	• May add meaningless ordinal relationship • Less interpretability of model parameters (feature level)

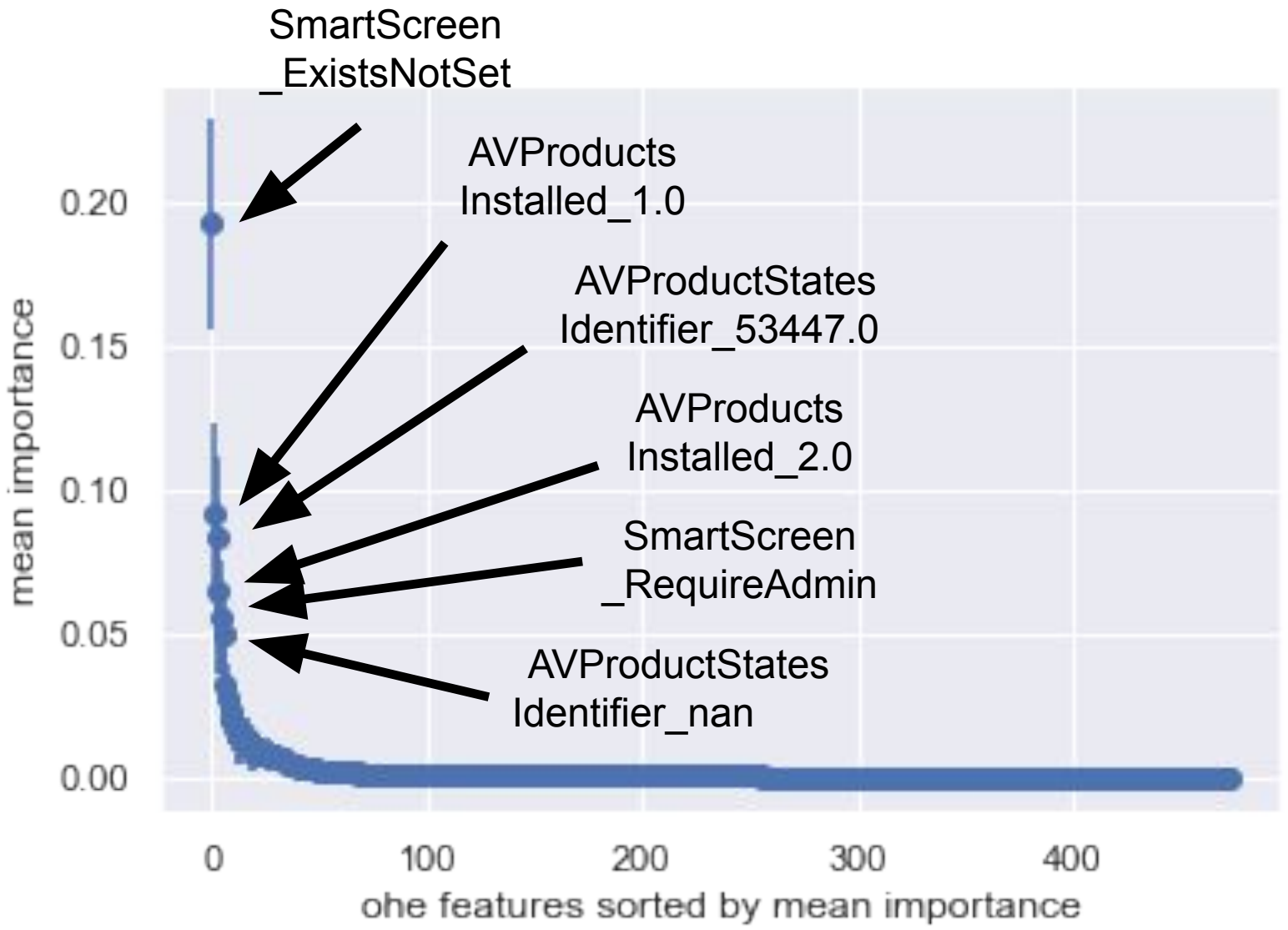
Getting a handle on the data - one-hot encoding (ohe) and filtering

Pre-processing Pipeline



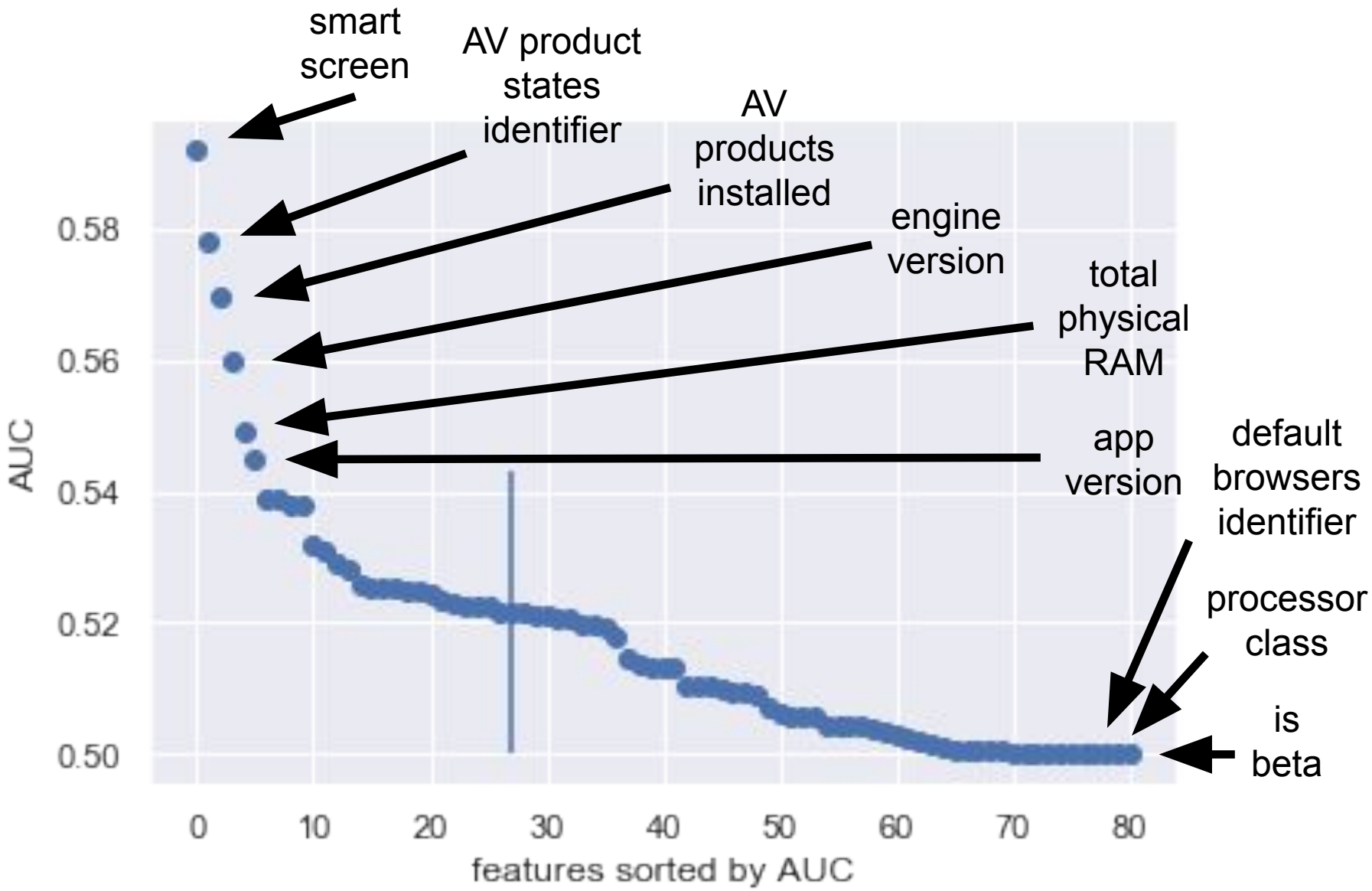
Which features are actually important?

Six ohe features are of clearly higher importance (Random Forest Classifier)



How far can you get by simply picking your favorite feature?

A single feature can score an AUC of up to ~0.59 (Logistic Regression Classifier)



How much better do you do by using more than one feature?

Logistic Regression and Random Forest perform similarly (AUC)

7 Fold Cross Validation

	Logistic Regression	Random Forest
One Feature	0.592	-
Top Three Features	0.639	0.639
Top Six Features	0.666	0.665
Top Twelve Features	**0.674**	0.664

Highest AUC achieved, time taken ~11 mins

deep trees – 0.673, however time taken ~58 mins

Kaggle Scores

Public Leader's AUC:
0.714

Private Leader's AUC:
0.675

What could be done next?

- Fine-tune hyperparameters of classifiers
- Use label encoding for ordered categorical features (e.g. version numbers)
- Engineer new features (e.g. extract timestamps from some feature labels)
- Employ more complex classifiers (e.g. neural networks)
- Keep in mind that ‘data leakage’ limits power of machine learning approaches

A reduced set of features can already yield powerful predictions

https://github.com/tkino15/kaggle_malware.git