Feature Engineering Workshop Review
And some more, advanced, really!

Recap

- **Supervised learning problem, multiclass**
- **200K lines of system logs.**
- **Source of each of these log lines need to be identified.**
- **17 variables.**
- **2 Classificational variables.**
- **11 Numerical variables.**
- **Data has been structured by an expert and a data scientist working together.**

Discussion

- **Check the result of the model.**
- **Error rate across the test set - 0.5%**

0	0.0001	2 / 19,279
0	1.0	1 / 1
0	0.0054	218 / 40,037

GENERAL

- Flow Web UI ...
- ... Importing Dat
- ... Building Mode
- ... Making Predict
- ... Using Flows
- ...Troubleshootin

Discussion

- **99.5% accurate ability to classify correctly.**
- **Great result?**

0	0.0001	2 / 19,279
0	1.0	1 / 1
0	0.0054	218 / 40,037

GENERAL

- Flow Web UI ...
- ... Importing Data
- ... Building Mode
- ... Making Predict
- ... Using Flows
- ...Troubleshootin

Discussion

- **99.5% accurate ability to classify correctly.**
- **Great result? No.**

0	0 / 118
0.1667	1 / 6
0.6316	144 / 228
0.6000	3 / 5
0	0 / 86
	0 / 0
0	0 / 191
0	0 / 7,169
0.5000	43 / 86
1.0	1 / 1

Discussion

- **The data is biased.**
- **The mass behaviour guides the class behaviour.**

0	0 / 118
0.1667	1 / 6
0.6316	144 / 228
0.6000	3 / 5
0	0 / 86
	0 / 0
0	0 / 191
0	0 / 7,169
0.5000	43 / 86
1.0	1 / 1

0	0 / 7,169
0.5000	43 / 86
1.0	1 / 1
0	0 / 964
	0 / 0
0.0003	2 / 7,349
0	0 / 136
	0 / 0
0	0 / 1,575
0	0 / 3
0	0 / 33
0.0122	4 / 329
0	0 / 58
0.0625	5 / 80
	0 / 0
0.6000	3 / 5
0	0 / 4
0.0001	2 / 19,279

Discussion

- **We can engineer more features. But..**
- **There is an issue of overfitting.**
- **We can brute force it by providing weights. But..**
- **Weights are only successful to some point.**

0	0 / 118
0.1667	1 / 6
0.6316	144 / 228
0.6000	3 / 5
0	0 / 86
	0 / 0
0	0 / 191
0	0 / 7,169
0.5000	43 / 86
1.0	1 / 1

0	0 / 7,169
0.5000	43 / 86
1.0	1 / 1
0	0 / 964
	0 / 0
0.0003	2 / 7,349
0	0 / 136
	0 / 0
0	0 / 1,575
0	0 / 3
0	0 / 33
0.0122	4 / 329
0	0 / 58
0.0625	5 / 80
	0 / 0
0.6000	3 / 5
0	0 / 4
0.0001	2 / 19,279

Discussion

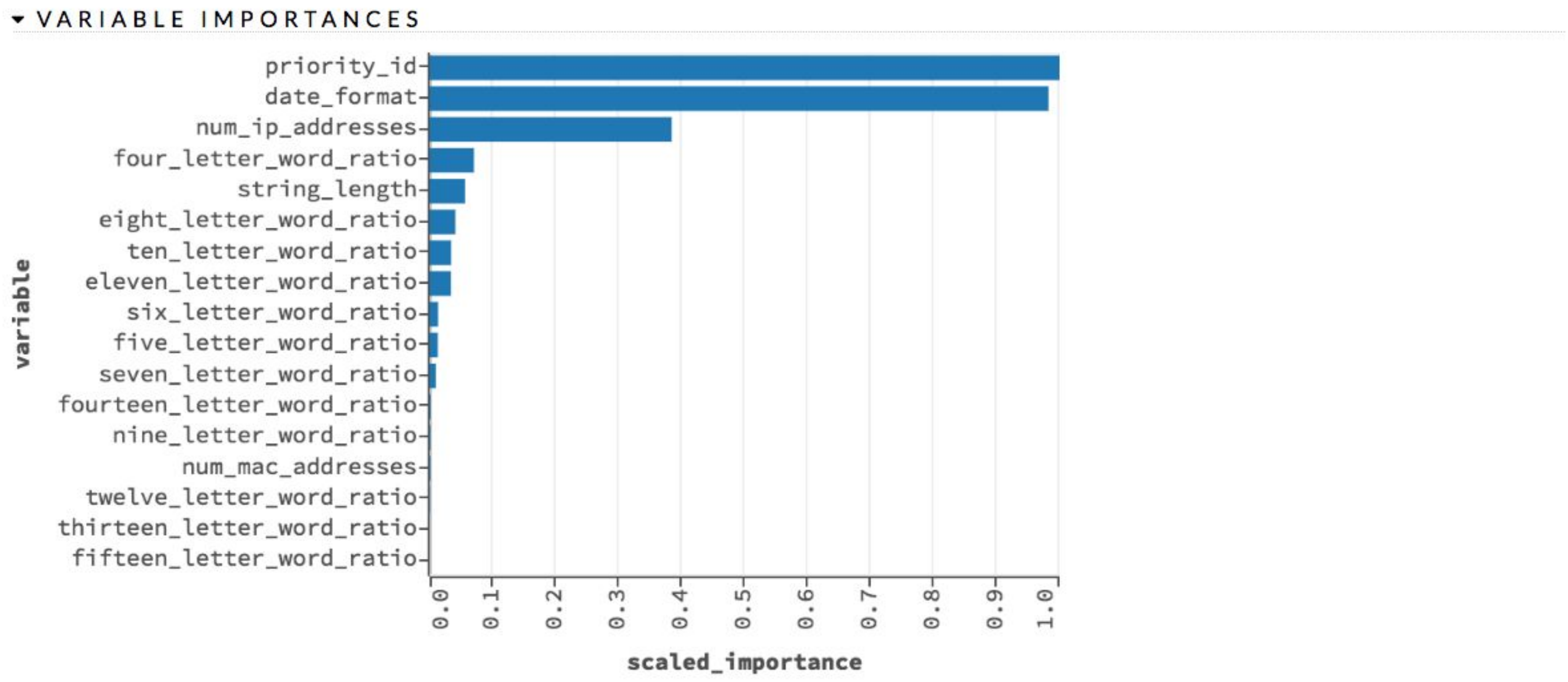
- **How do you provide weight to the features, but not add weights?**
- **Therefore you have to do some analysis on the features.**

Discussion

- **Therefore you have to do some analysis on the features.**
- **How?**

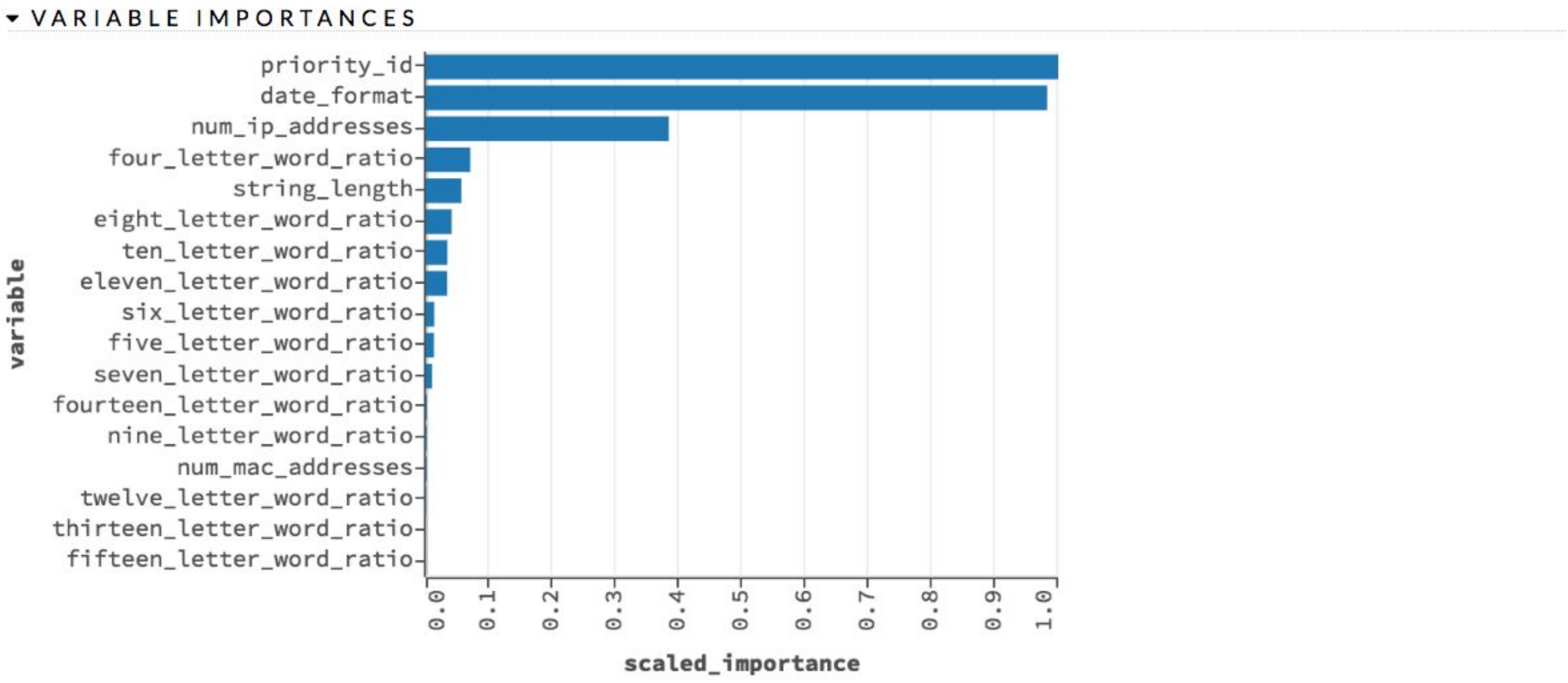
Discussion

- **Therefore you have to do some analysis on the features.**
- **How?**
- **Variable importance.**



Discussion

- **An influential feature could also be a disrupting feature.**
- **Especially in biased data. - Date Format**



Discussion

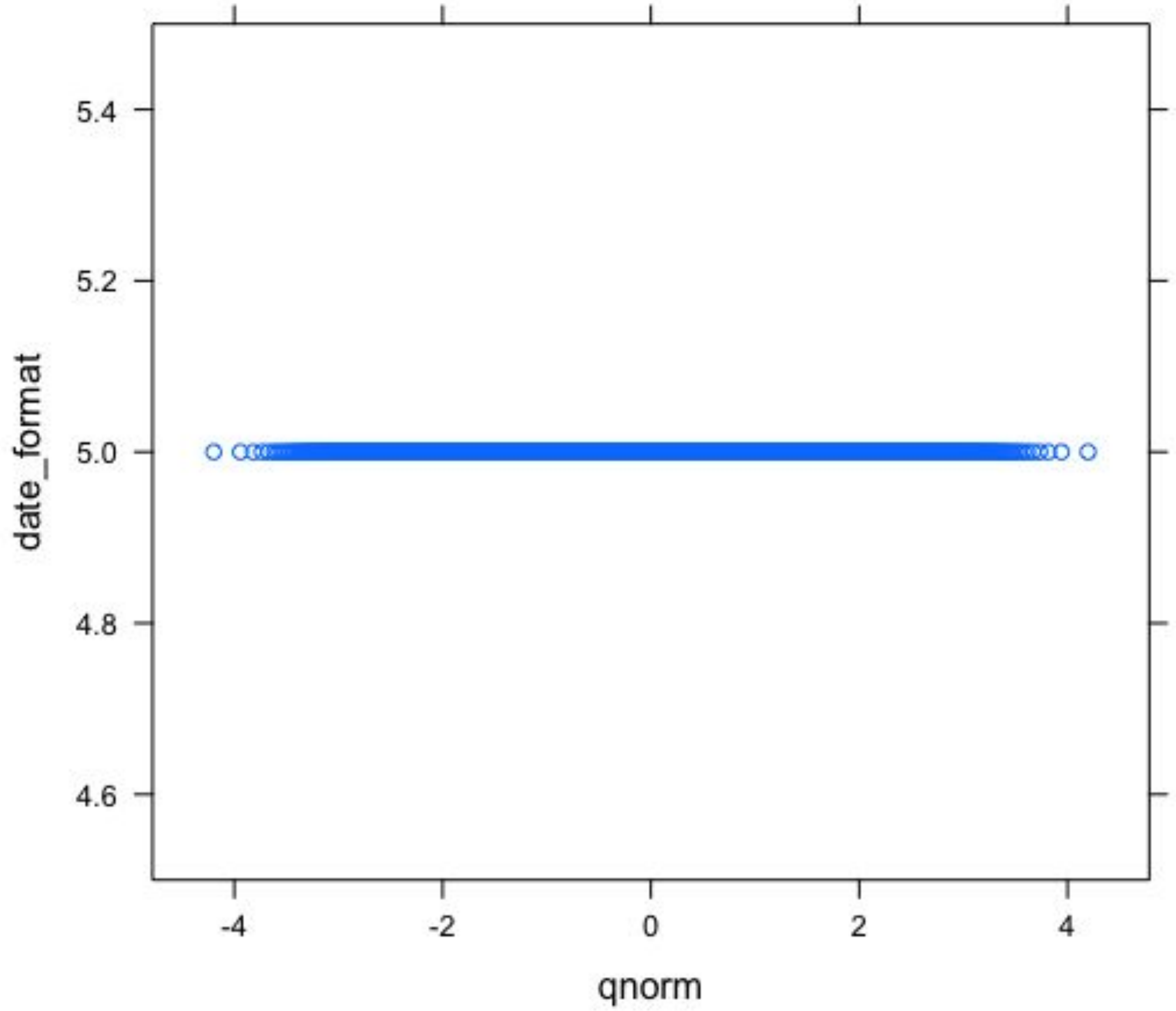
- **Pick two accurate classes interacting with a not so accurate class.**
- **And do a date format analysis on it.**

Discussion

- **Why did we choose date format as my target for analysis?**
- **Categorical -> meaning, not continuous and is a grouping variable!**

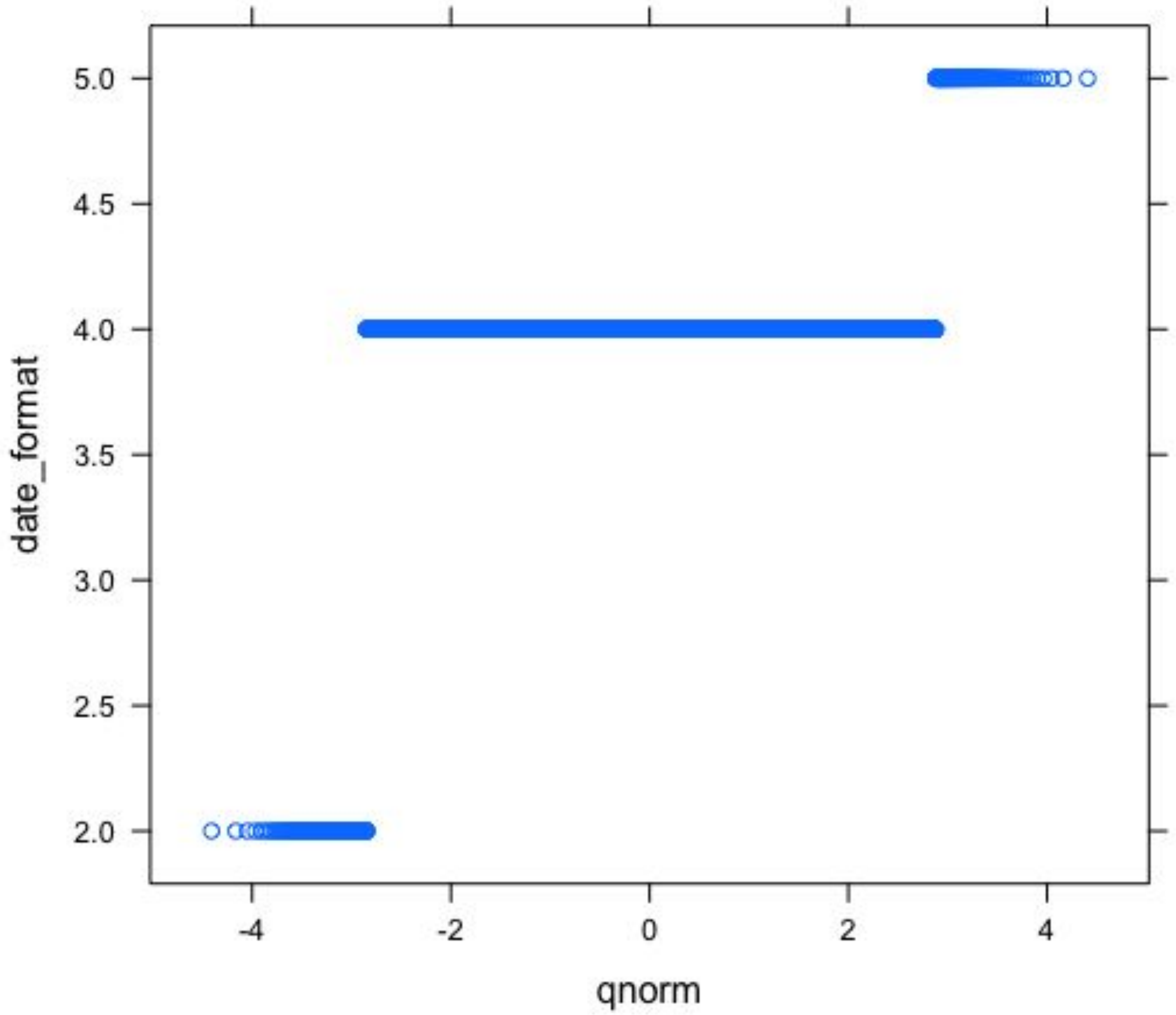
Discussion

- **Distribution of date from network.txt class**



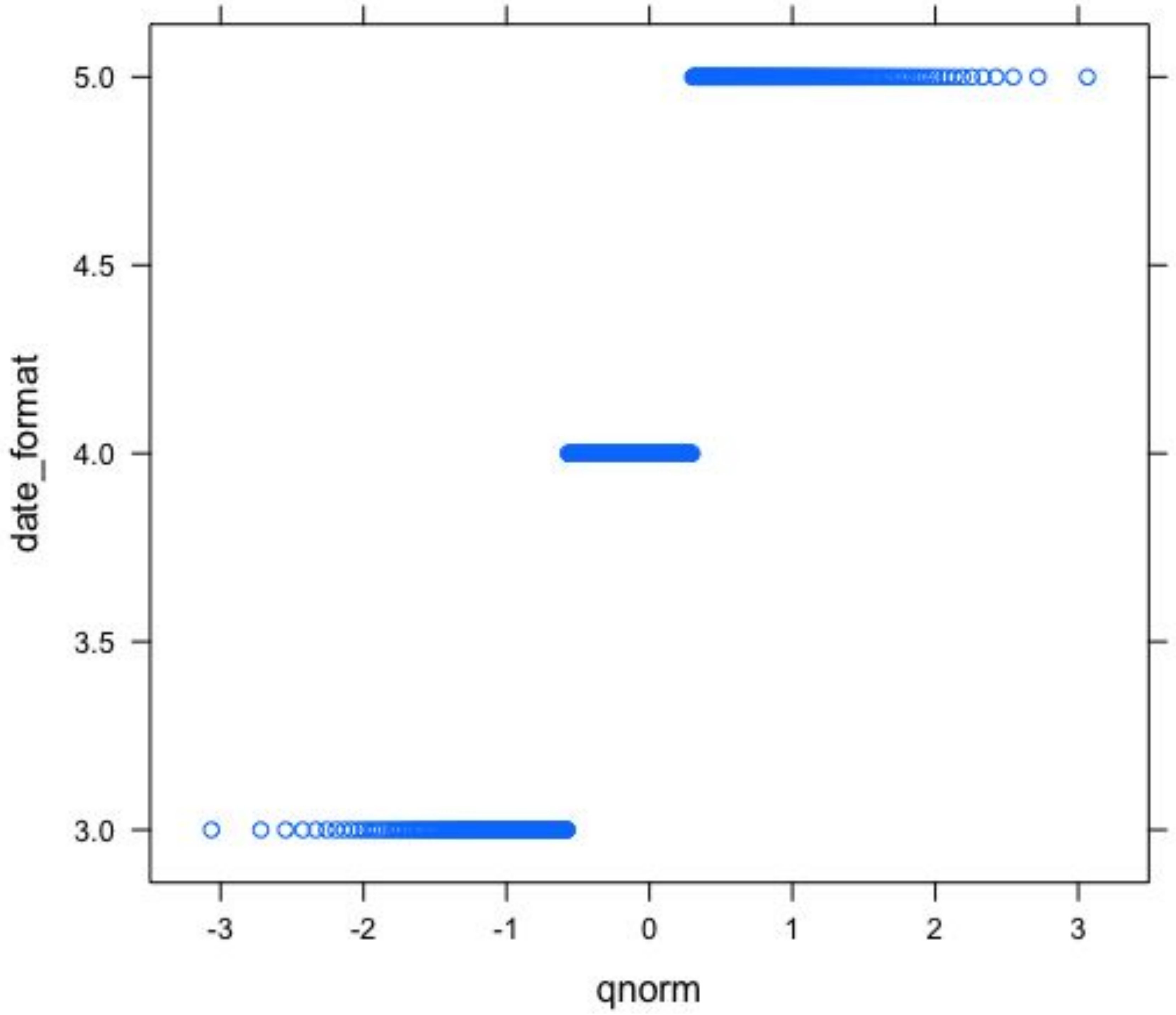
Discussion

- **Distribution of date from vmware.txt class**



Discussion

- **Distribution of date from vmware.txt class**



Discussion

- **Aggregate Encoding**
- **Encoding a feature in a class to its majority value to provide some weights.**

Discussion

- **Theoretically it is very easy.**
- **Practically YOU ARE CHANGING the truth.**
- **When you are implementing this you need the “source of data” to know this translation.**

Stacking

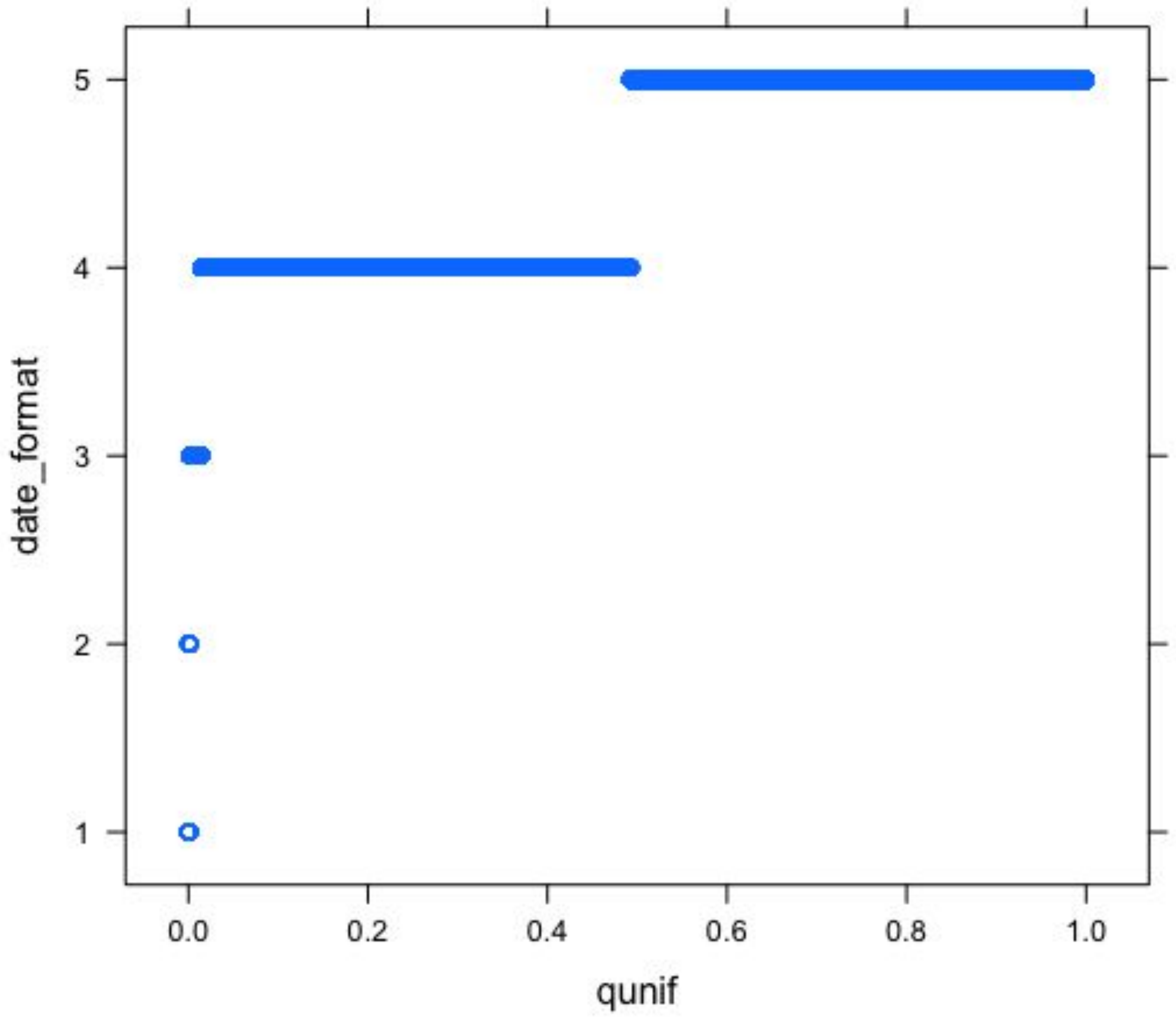
- **The process of dividing and recombining the subsets of results.**
- **Each subset is a meta-model.**

Stacking

- **You can divide the data by each class.**
- **Or, divide the data by certain distribution of a feature and then divide it.**

Stacking

- **Dividing the data by distribution**



Stacking

- **Advantages**
- **Stacking is incredibly powerful as it aggregates results from different model. But, robustly.**
- **Robustly? Yes, it ignores overfitting aspects of a model and regresses neatly to the mean.**

Re-Stacking/AutoStacking

- **Stacking is the idea of having multiple meta learners that feed into a larger model.**
- **What is Re-Stacking? Or Auto Stacking**

Re-Stacking/AutoStacking

- **The idea here is to feed the output of the model back to itself.**

Re-Stacking/AutoStacking

- **Why feed the model back to itself?**
- **The outcome of the model provides more value to the new model, and also makes it robust.**

Re-Stacking/AutoStacking

- **Any pitfalls of Autostacking?**
- **Yes, of course!**
- **Do it only after all hope is lost!**
- **It will give you small increments, but that might be the defining increment between governance accepting or rejecting your model.**

Re-Stacking/AutoStacking

- **How does it look after engineering?**
- **Now with 56 explanatory variables! All Brand new (not!)**

label	type	Miss
priority_id	enum	
date_format	enum	
string_length	int	
num_ip_addresses	int	
num_mac_addresses	int	
log_class	enum	
four_letter_word_ratio	real	
five_letter_word_ratio	real	
six_letter_word_ratio	real	
seven_letter_word_ratio	real	
eight_letter_word_ratio	real	
nine_letter_word_ratio	real	
ten_letter_word_ratio	real	
eleven_letter_word_ratio	real	
twelve_letter_word_ratio	real	
thirteen_letter_word_ratio	real	
fourteen_letter_word_ratio	real	
fifteen_letter_word_ratio	real	
predict	enum	
airmagnet.txt	real	

COLUMN SUMMARIES

label	type	Missing	Ze
arubanetworks.txt	real	0	
bigip.vpn.txt	real	0	
bluecoat.txt	real	0	
bugreport.txt	real	0	
centrify.txt	real	0	
checkpoint.txt	real	0	
ciscoacl.txt	real	0	
ciscoacs.txt	real	0	
ciscoasa.txt	real	0	
cluster_manager.txt	real	0	
clusterd.txt	real	0	
cyberark.txt	real	0	
default.txt	real	0	
esxad.txt	real	0	
f5.txt	real	0	
ftp.txt	real	0	
hpnetwork.txt	real	0	
incidentserver.txt	real	0	
infoblox.txt	real	0	
ironport.txt	real	0	

COLUMN SUMMARIES

label	type	Missir
loggagg.txt	real	
mail.txt	real	
mesosphere.txt	real	
mocana.txt	real	
network.txt	real	
networkadmin.txt	real	
oracle.txt	real	
paloalto.txt	real	
postgres.txt	real	
radware.txt	real	
ssh.txt	real	
stunnel.txt	real	
system.txt	real	
trendmicro.txt	real	
uiserver.txt	real	
unix_system.txt	real	
vmware.txt	real	
xinetd.txt	real	

Re-Stacking/AutoStacking

- **Example of output before and after autostacking**
- **Difference of 0.08%**

Before

0	0.0001	2 / 19,279
0	1.0	1 / 1
0	0.0054	218 / 40,037

Flow packs are a gre
learn H_2O. Try out t
in your browser.
Browse installed pa

H_2O REST API

- Routes

After

0	0.0001	2 / 19,279
1	0	0 / 1
1	0.0046	183 / 40,037

in your browser.
Browse installed pac

H_2O REST API

- Routes

Re-Stacking/AutoStacking

- **What about all the Multi-class unbiased-ness you were talking about?**
  - **Before**

0.1667	1 / 6
0.6316	144 / 228
0.6000	3 / 5
0	0 / 86
	0 / 0
0	0 / 191
0	0 / 7,169
0.5000	43 / 86
1.0	1 / 1

  - **After**

0	0 / 6
0.6009	137 / 228
0	0 / 5
0	0 / 86
0	0 / 191
0	0 / 7,169
0.4651	40 / 86
1.0	1 / 1

Tooling

- **Visualisations in H2O**
- **Variable Importance**
- **Helps identify class distribution**
- **Confusion Matrix**
- **Helps quickly autostack.**
- **I actually use a lot of flow and R lattice while analysing.**
- **My tooling helps me to be fast!**

Thank You

Questions?