Game of Fraud Detection with SQL and Machine Learning

Chris Saxon, Oracle & Abi Giles-Haigh, Chief Data Science Officer

ORACLE Platinum Partner

The Iron Bank is

Going Bust!

Photo by Katie Harp on Unsplash


We need new ways
to solve an old problem…


SQL guy
@ChrisRSaxon


Data Science Girl
@Abi_Giles_Haigh

The Hand of the King analysed fraud trxns

Causes of fraud

1. Value = 10M
2. Same value transfer from/to the same house

SQL

```
desc got_transactions

Name                          Null? Type
----------------------------- ----- --------------
```

Vertice ORACLE Platinum Partner

SQL

TRANSACTION_CATEGORY	VARCHAR2(26)
TRANSACTION_AMOUNT	NUMBER(38,2)
TRANSACTION_TYPE	VARCHAR2(26)
TRANSACTION_DATE	DATE
TRANS_ID	NUMBER(10)
FRAUD	NUMBER(38)

SQL

```
SENDER_ID                VARCHAR2(26)
SENDER_NAME              VARCHAR2(100)
SENDER_HOUSE             VARCHAR2(100)
SENDER_GENDER            VARCHAR2(26)
SENDER_NOBILITY          VARCHAR2(26)
SENDER_HOME_OWNER        VARCHAR2(26)
SENDER_INCOME            NUMBER(38)
```

SQL

```
RECEIVER_ID                VARCHAR2(26)
RECEIVER_NAME              VARCHAR2(100)
RECEIVER_HOUSE             VARCHAR2(100)
RECEIVER_GENDER            VARCHAR2(26)
RECEIVER_NOBILITY          VARCHAR2(26)
RECEIVER_HOME_OWNER        VARCHAR2(26)
RECEIVER_INCOME            NUMBER(38)
```

SQL

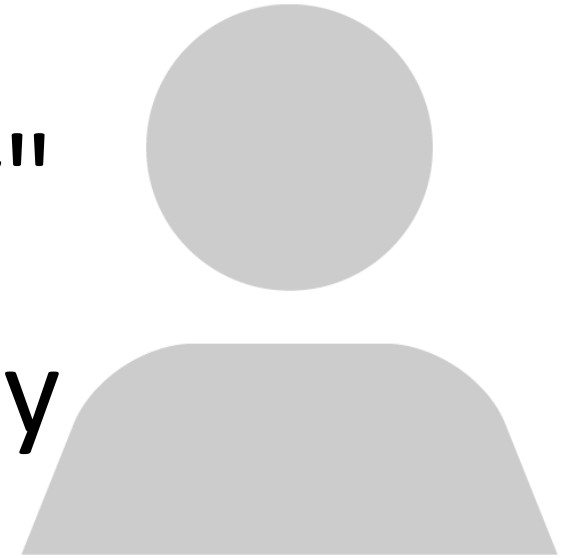
TDT	TAMOUNT	TTYPE	SENDER_HOUSE	RECEIVER_HOUSE
04-JAN-2017	257,657.85	CASH_IN	House Tyrell	House Lannister
03-FEB-2017	105,861.07	TRANSFER	House Arryn	House Stark
06-FEB-2017	2,410.38	DEBIT	House Tyrell	House Baratheon
20-FEB-2017	322.94	PAYMENT	House Arryn	House Stark
21-FEB-2017	1,559.41	CASH_OUT	House Tyrell	House Targaryen
04-MAY-2017	1,624.57	CASH_IN	House Arryn	House Baratheon
07-JUN-2017	2,070.47	CASH_OUT	Night's Watch	House Arryn
22-JUN-2017	3,392.80	DEBIT	House Tully	House Arryn
03-JUL-2017	1,767.72	PAYMENT	Wildling	House Lannister
16-JUL-2017	56,125.55	TRANSFER	House Greyjoy	House Stark

Rule 1

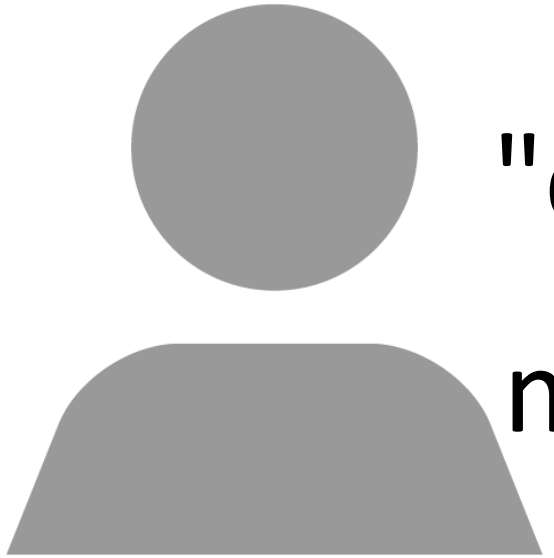
```
select t.*
from   got_transactions t
where  transaction_amount = 10000000
```

Rule 2

"dirty" money

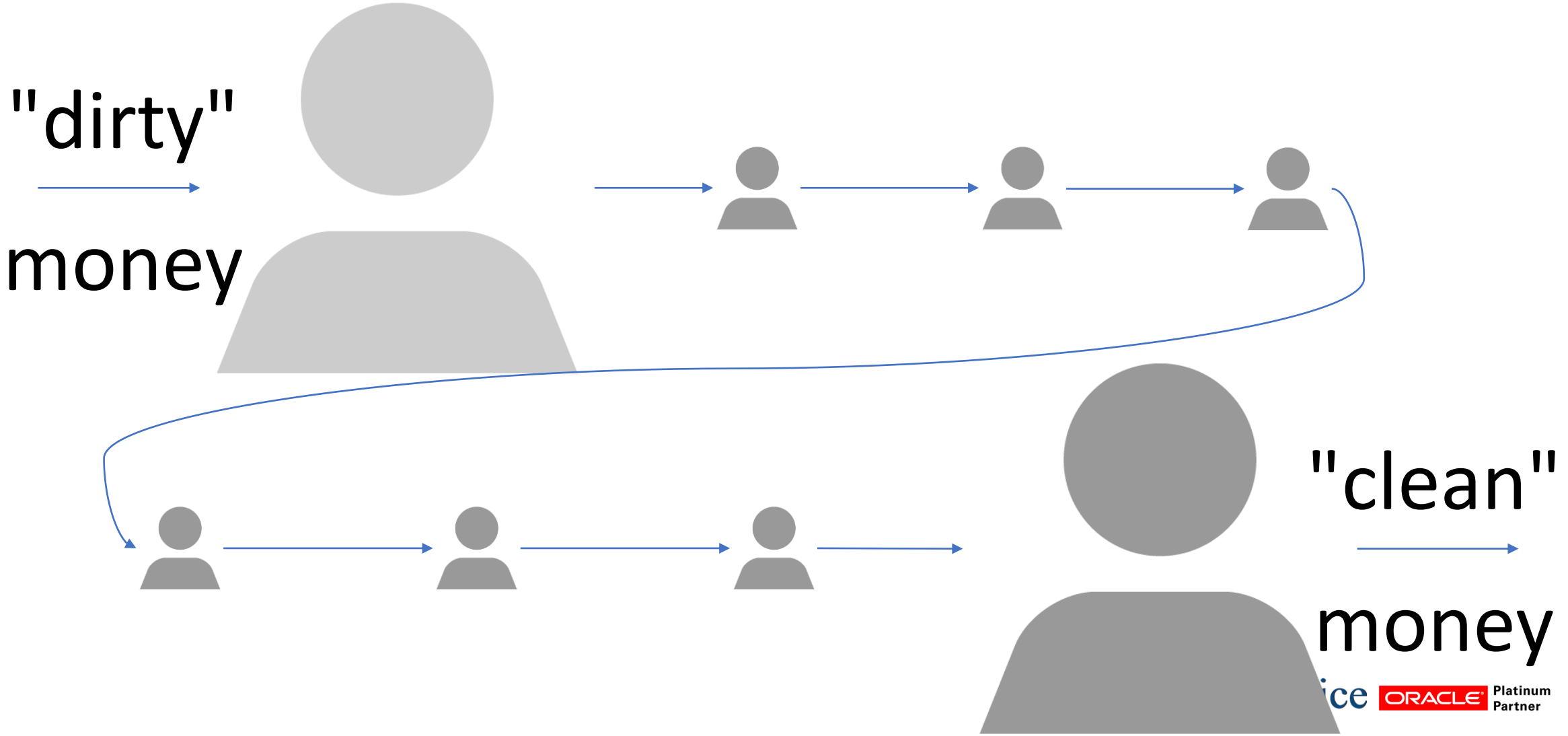


transfer



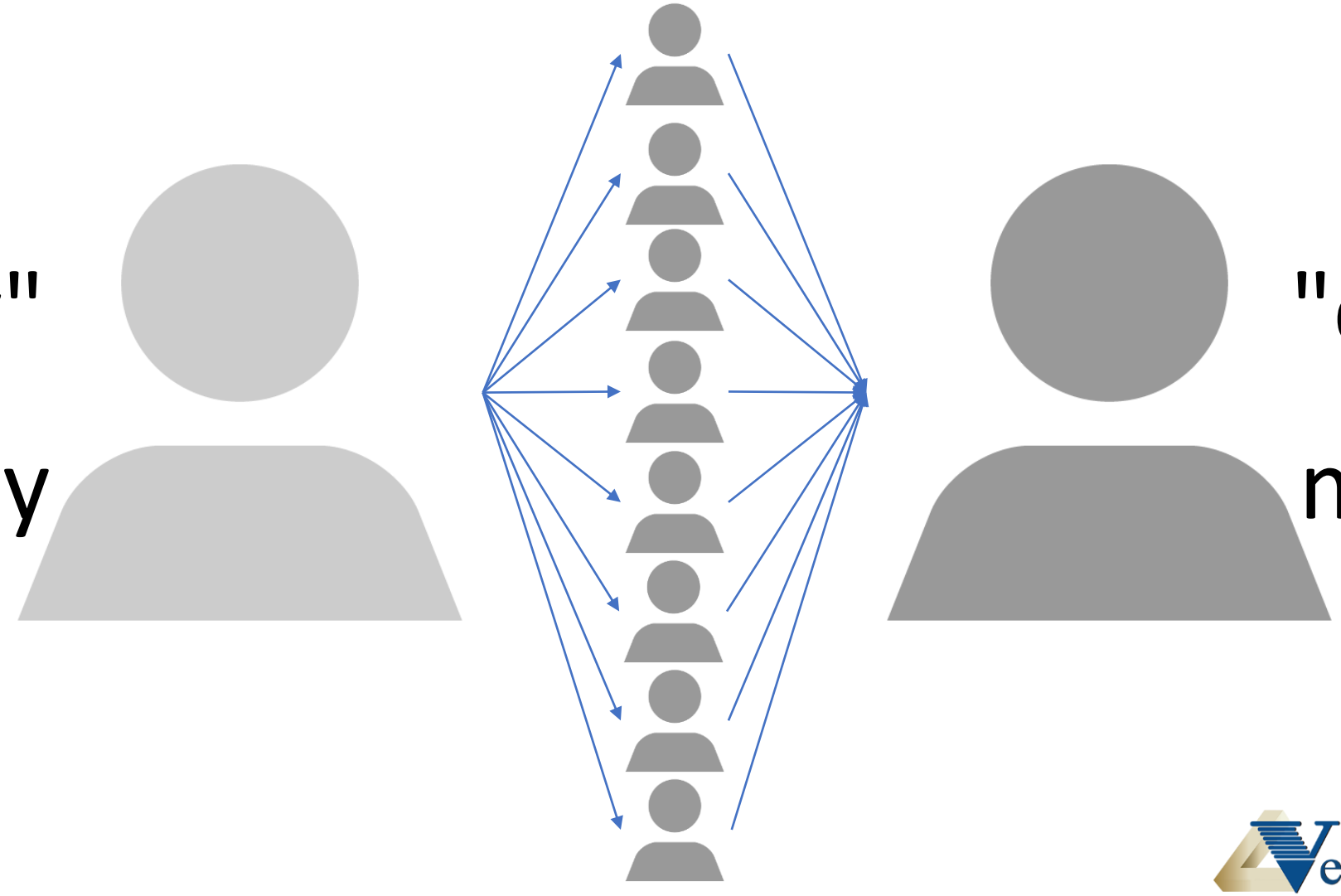
"clean" money

Of course the real world is more like…



or...

"dirty" money



"clean" money

SQL

```
case
  when transaction_type in (
    'CASH_IN', 'TRANSFER', 'CASH_OUT'
  ) and max ( trans_id ) over ( … ) is not null
    then 1 -- FRAUD, it's FRAUD!
    else 0
end
```

Vertice ORACLE Platinum Partner

SQL

```
case
  when transaction_type in (
    'CASH_IN', 'TRANSFER', 'CASH_OUT'
  ) and max ( trans_id ) over ( … ) is not null
    then 1 -- FRAUD, it's FRAUD!
    else 0
end
```

Vertice ORACLE Platinum Partner

SQL

```
max ( trans_id ) over (
   partition by sender_house, receiver_house,
                transaction_amount




)
```

SQL

```
max ( trans_id ) over (
  partition by sender_house, receiver_house,
               transaction_amount
  order by transaction_date


)
```

SQL

```
max ( trans_id ) over (
   partition by sender_house, receiver_house,
                transaction_amount
   order by transaction_date
   range between unbounded preceding and
         current row
)
```

All previous and itself!

SQL

```
max ( trans_id ) over (
   partition by sender_house, receiver_house,
                transaction_amount
   order by transaction_date
   range between 90 preceding and
         1 preceding
)
```

Excludes today…

SQL

```
max ( trans_id ) over (
   partition by sender_house, receiver_house,
                transaction_amount
   order by transaction_date
   range between 90 preceding and
         1/1440 preceding
)
```

Previous *minute*

Multiple transfers same amount

```
count (*) over (
  partition by sender_house, receiver_house,
               transaction_amount
  order by transaction_date
  range between 90 preceding and
        1/1440 preceding
)
```

So how successful are we?

Error rate ~1%

Fraud rate ~1%

We did it!

Vertice ORACLE Platinum Partner

SENDER_HOUSE	RECEIVER_HOUSE	TRX#
None	None	321
Night's Watch	Night's Watch	216
House Greyjoy	House Greyjoy	73
House Baratheon	House Baratheon	71
House Lannister	House Lannister	57

https://gameofthrones.fandom.com/wiki/Jon_Snow/

But...

We're *still* losing money

Image by 1820796 from Pixabay

… and the Baratheons are complaining their trxns are blocked

WTF?!

Theory

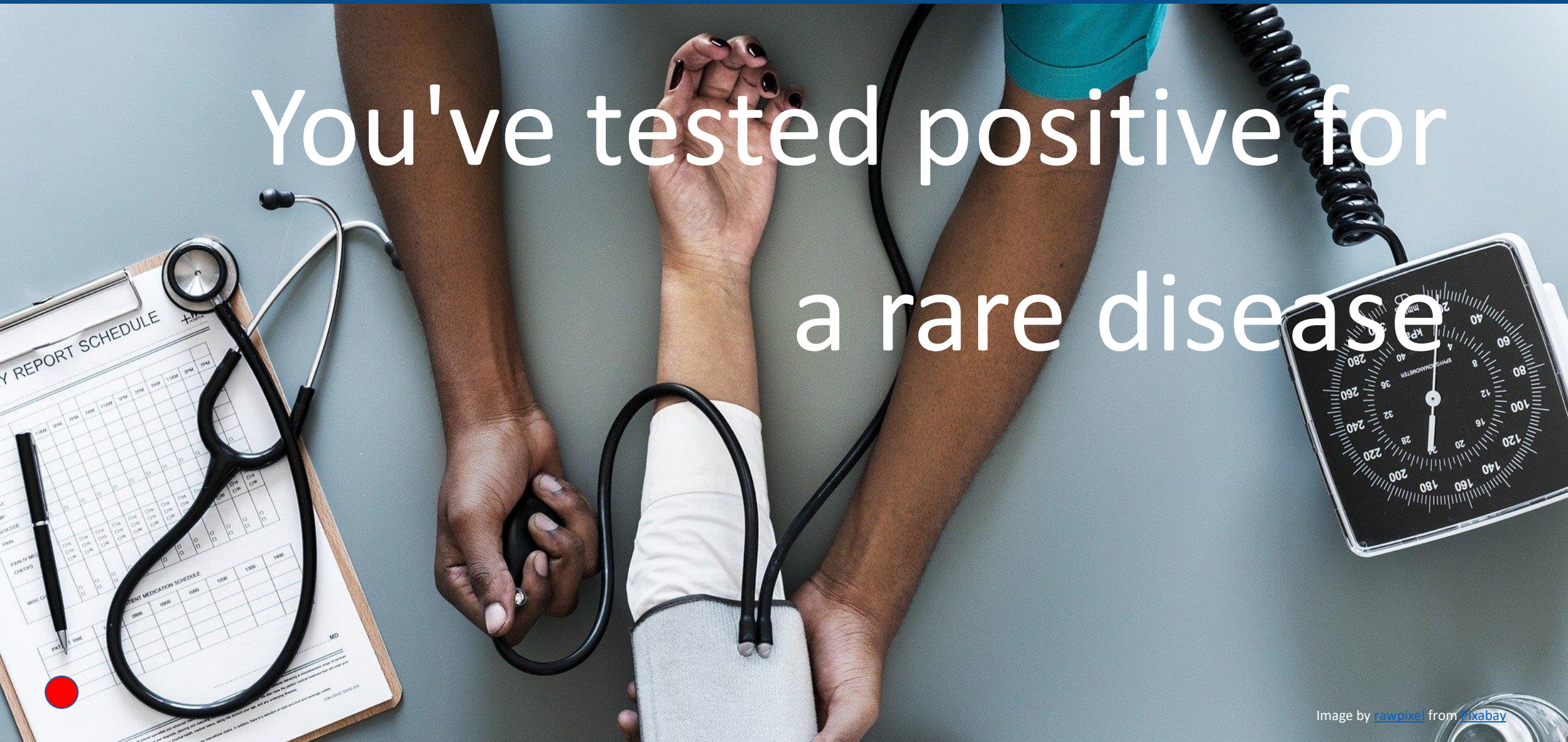


Image by rawpixel from Pixabay

Theory

It affects ~ 1 in 10,000 people

Theory

The test is

~ 99% accurate

Theory

Are you scared?

		Reality	
		Sick (100)	Healthy (999,900)
Test Results	Sick	99	
	Healthy		989,901

		Reality	
		Sick (100)	Healthy (999,900)
Test Results	Sick	99	
	Healthy	1	989,901

		Reality	
		Sick (100)	Healthy (999,900)
Test Results	Sick	99 ← **100x**	9,999
	Healthy	1	989,901

Classifying events: confusion matrix

Confusion Matrix

		Actual Values	
		Positive (1)	**Negative (0)**
Predicted Values	**Positive (1)**	True positive	False positive
	Negative (0)	False negative	True negative

So how do our rules do?

Vertice ORACLE Platinum Partner

SQL: Results

Error rate ~1%

Mark everything "**NOT FRAUD**"!

Fraud rate ~1%

Vertice ORACLE Platinum Partner

Confusion Matrix

		Actual Values	
		Positive (1)	**Negative (0)**
Predicted Values	**Positive (1)**	80	
	Negative (0)		115,519

Confusion Matrix

		Actual Values	
		Positive (1)	**Negative (0)**
Predicted Values	**Positive (1)**	80 ← **15x**	1,203
	Negative (0)	**10x** ↑ 872	115,519

Vertice ORACLE Platinum Partner

So how can we do better?

Vertice ORACLE Platinum Partner

Machine Learning!

Machine Learning

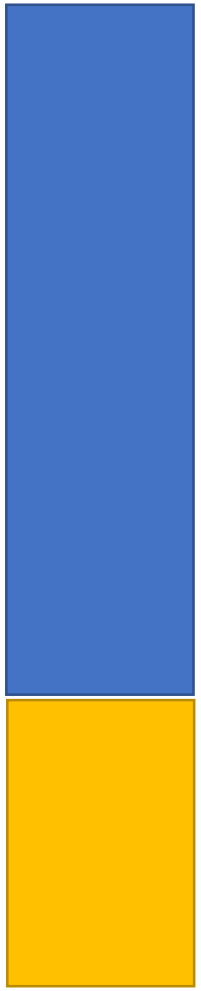
Supervised: Labels ☺

Unsupervised: No labels ☹

SQL

TDT	TAMOUNT	TTYPE	SENDER_HOUSE	RECEIVER_HOUSE	FRAUD
04-JAN-2017	257,657.85	CASH_IN	House Tyrell	House Lannister	1
03-FEB-2017	105,861.07	TRANSFER	House Arryn	House Stark	0
06-FEB-2017	2,410.38	DEBIT	House Tyrell	House Baratheon	0
20-FEB-2017	322.94	PAYMENT	House Arryn	House Stark	0
21-FEB-2017	1,559.41	CASH_OUT	House Tyrell	House Targaryen	0
04-MAY-2017	1,624.57	CASH_IN	House Arryn	House Baratheon	0
07-JUN-2017	2,070.47	CASH_OUT	Night's Watch	House Arryn	1
22-JUN-2017	3,392.80	DEBIT	House Tully	House Arryn	0
03-JUL-2017	1,767.72	PAYMENT	Wildling	House Lannister	0
16-JUL-2017	56,125.55	TRANSFER	House Greyjoy	House Stark	0

Machine Learning: Split the data!



Train Data (80%)

```
got_transactions SAMPLE(80)
seed(124)
```

Test Data (20%)

… **Minus** …

Supervised Modeling: Settings

Step 2: Setting for the algorithm…..

```
INSERT INTO MODEL_SETTINGS (setting_name, setting_value)
        VALUES ('ALGO_NAME', 'ALGO_NAIVE_BAYES');

INSERT INTO MODEL_SETTINGS (setting_name, setting_value)
        VALUES ('PREP_AUTO', 'ON');
```

Vertice ORACLE Platinum Partn

Supervised Modeling: Modeling

Step 3: Build the model….

```
DBMS_DATA_MINING.CREATE_MODEL
```

Supervised Modeling: Modeling

Step 2: Build the model….

```
model_name                => 'GOT_FRAUD_MODEL',
mining_function           => dbms_data_mining.classification,
data_table_name           => 'got_transactions_train',
case_id_column_name       => 'trans_id',
target_column_name        => 'fraud',
settings_table_name       => 'MODEL_SETTINGS',
);
```

Supervised Modeling: Apply the model

Step 4: Apply the model….

DBMS_DATA_MINING.APPLY

Supervised Learning: Apply the model

Step 4: Apply the model….

```
model_name                => 'GOT_FRAUD_MODEL',
data_table_name           => 'got_transactions_test',
case_id_column_name       => 'trans_id',
results_table_name     =>   'FRAUD_RESULTS',
);
```

Vertice ORACLE Platinum Partner

ML: Results

Error rate ~3%

Fraud rate ~1%

Worse than rules?!

Create the confusion!

DBMS_DATA_MINING.COMPUTE_CONFUSION_MATRIX

The code…

```
accuracy                          => v_accuracy,
apply_result_table_name           => 'FRAUD_NB_RESULTS',
target_table_name                 => 'got_transactions_test',
case_id_column_name               => 'trans_id',
target_column_name                => 'FRAUD',
confusion_matrix_table_name       => 'Fraud_NB_confusion_matrix',
score_column_name                 => 'PREDICTION',
score_criterion_column_name       => 'COST',
score_criterion_type              => 'COST'
  );
```

Vertice ORACLE Platinum Partn

Confusion Matrix

		Actual Values	
		Positive (1)	**Negative (0)**
Predicted Values	**Positive (1)**	832	4,566
	Negative (0)	451	111,825

3x

0.5x

SQL

TRANS_ID	PREDICTION	PROBABILITY
731418	0	0.999999999985
731418	1	0.000000000015
1010855	0	0.999999999998
1010855	1	0.000000000002
1013297	0	1.000000000000
1013297	1	0.000000000000
1153504	0	0.999999999496
1153504	1	0.000000000504

Vertice ORACLE Platinum Partner

```
select trans_id,
       row_number () over (
         partition by trans_id
         order by probability desc
       ) rn
from   fraud_nb_results
```

Vertice ORACLE Platinum Partner

```
with ranks as (
   select trans_id,
          row_number () over (
            partition by trans_id
            order by prediction desc
          ) rn
   from   fraud_nb_results
)
   select * from ranks where rn = 1
```

Vertice ORACLE Platinum Partner

SENDER_HOUSE	RECEIVER_HOUSE	TRX#
House Lannister	House Lannister	421
House Arryn	House Arryn	41
Night's Watch	Night's Watch	38
House Greyjoy	House Greyjoy	35
None	None	28

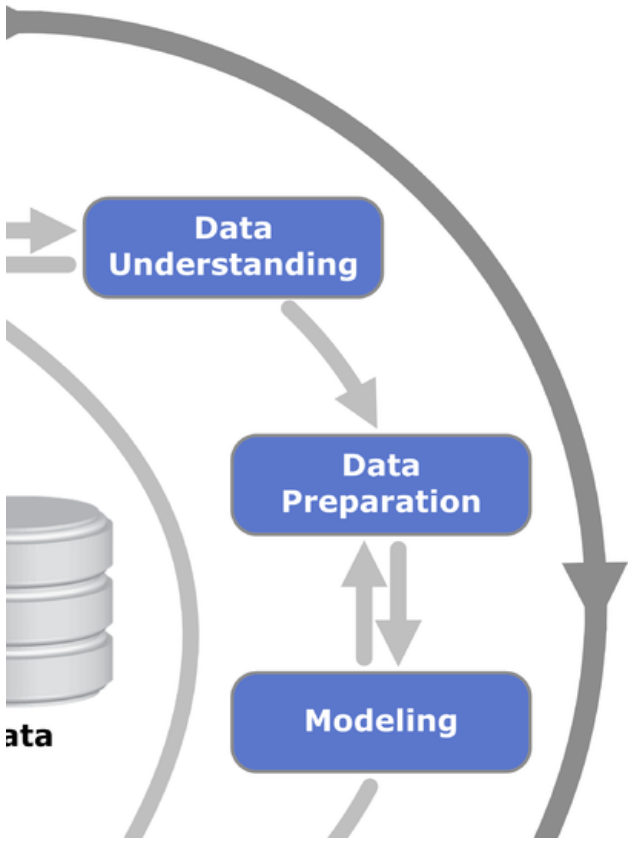
It's the Lannisters!

https://guff.com/this-theory-about-the-lannister-siblings-changes-everything-we-thought-we-knew-about-game-of-thrones

Can we reduce the errors?

Combine SQL & ML!

Data Science Approach



SQL Rules

+

Machine Learning

=

The seven kingdoms….

Vertice ORACLE Platinum Partn

Join Rules & ML

```
select gt.trans_id,
       greatest (
         ml.fraud_flag, gt.fraud_flag
       ) fraud_flag
from   got_transactions_rules_test gt
join   got_transactions_ml_test_results ml
on     gt.trans_id = ml.trans_id
```

Vertice ORACLE Platinum Partner

Confusion Matrix

		Actual Values	
		Positive (1)	**Negative (0)**
Predicted Values	**Positive (1)**	898	5,073
	Negative (0)	385	111,318

5x

0.4x

Vertice ORACLE Platinum Partner

Confusion Matrix: Final comparison

		Actual Values	
		Positive (1)	**Negative (0)**
Predicted Values	**Positive (1)**	SQL: 80 ML: 832 **SQL + ML: 898**	**SQL: 1,203** ML: 4,566 SQL + ML: 5,073
	Negative (0)	SQL: 872 ML: 451 **SQL + ML: 385**	**SQL: 115,519** ML: 111,825 SQL + ML: 111,318

Vertice ORACLE Platinum Partner

Deterministic vs Probabilistic

Vertice ORACLE Platinum Partner

Deterministic

if <rule 1> then …

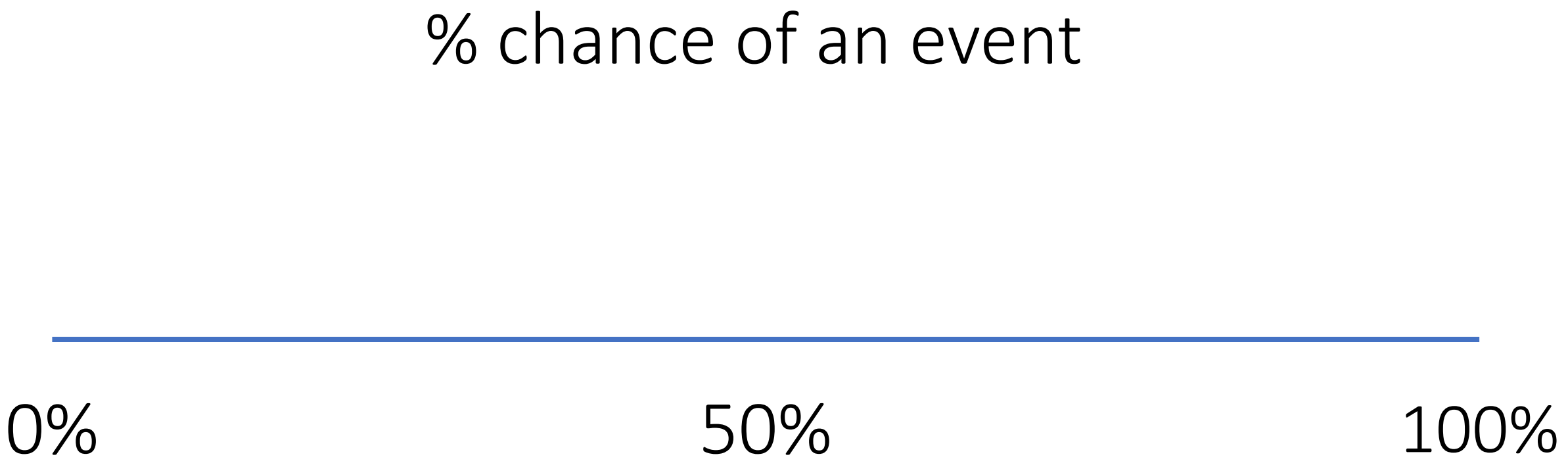
if <rule 2> then …

if <rule 3> then …

else …

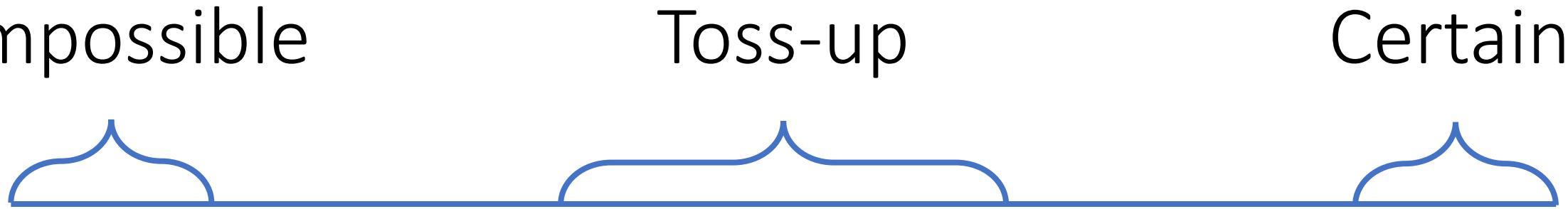
Vertice ORACLE Platinum Partner

How most people view probabilities

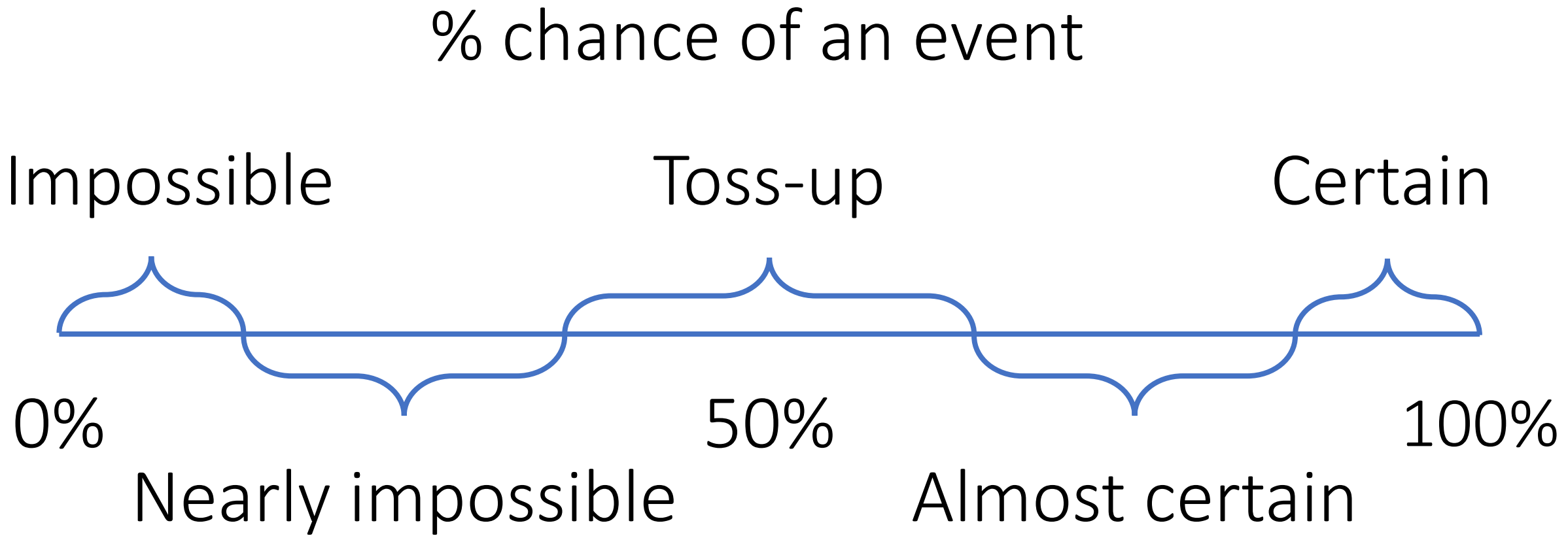


How most people view probabilities

% chance of an event



How most people view probabilities



SQL rules vs ML

- Deterministic -> guaranteed to flag; easy to test/verify
- Needs to be told rules
  - Need domain knowledge
- Existing skills
- You KNOW why trx flagged

- Probabilistic -> don't know the outcome
- Can find rules
  - Might miss "obvious" fraud
- New skills & approaches
- Why flagged? ¯_(ツ)_/¯

Vertice ORACLE Platinum Partner

The real world

- What's the impact of false positive & false negative?
- How much effort is it to resolve misclassification?
- How high is the false positive rate?
- How do explain the results if someone wants to know?

Which one are you ready for?

Vertice ORACLE Platinum Partn

Q&A

@VerticeCloud
Connect@VerticeCloud.com

SQL Guy: Chris Saxon @ChrisRSaxon

ML Girl: Abi Giles-Haigh @Abi_Giles_Haigh

www.verticecloud.com