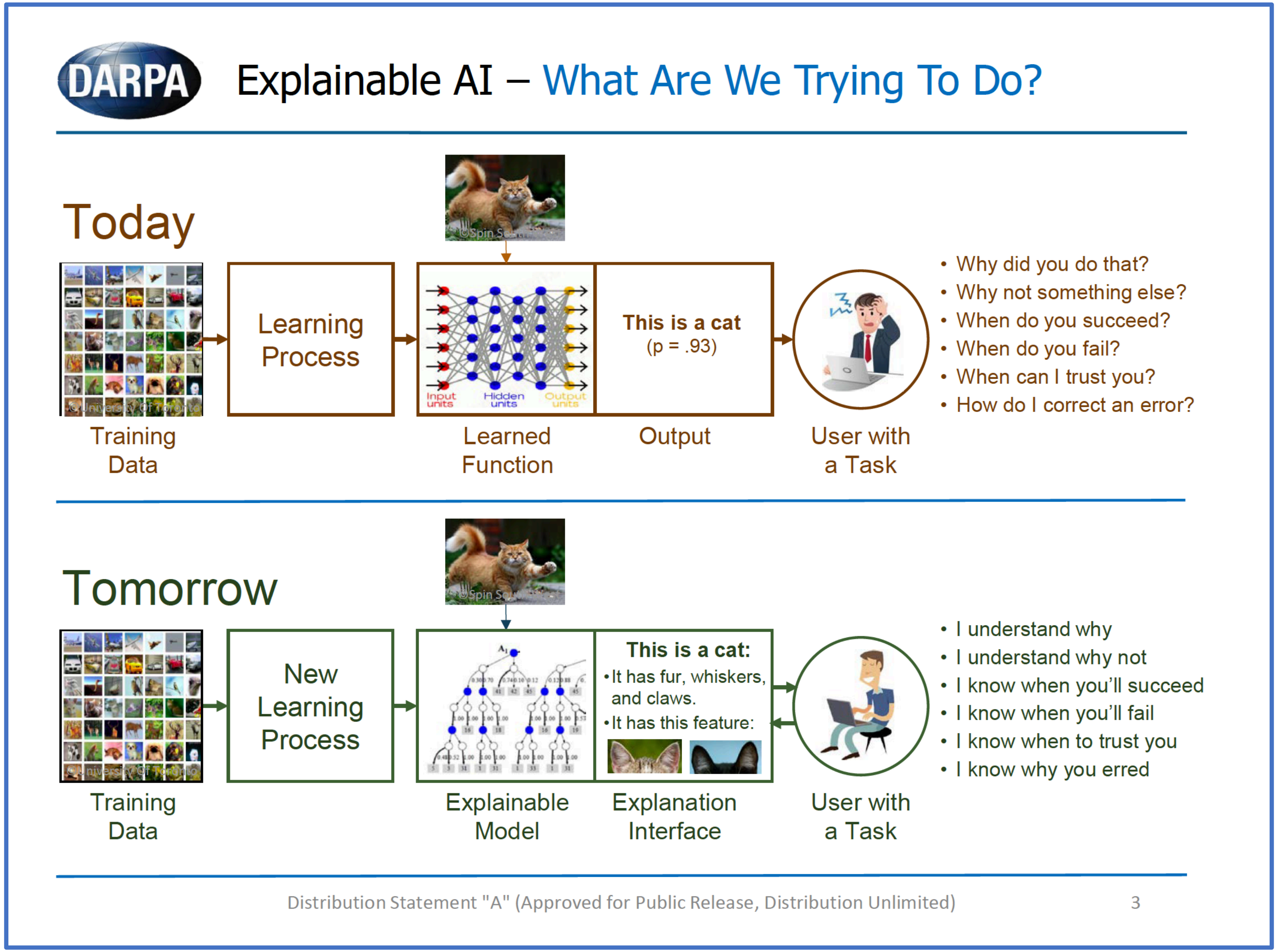
Explainable AI

Rick Kuhn, CSD and Raghu Kacker, ACMD

What is the problem?

- Artificial intelligence and machine learning (AI/ML) systems have exceeded human performance in nearly every application where they have been tried
- AI is starting to be incorporated into consumer products. This trend is accelerating, and AI will be increasingly used in safety-critical systems
- AI systems are good, but sometimes make mistakes, and human users will not trust their decisions without explanation
- There is a tradeoff between AI accuracy and explainability: the most accurate methods, such as convolutional neural nets (CNNs), provide no explanations; understandable methods, such as rule-based, tend to be less accurate

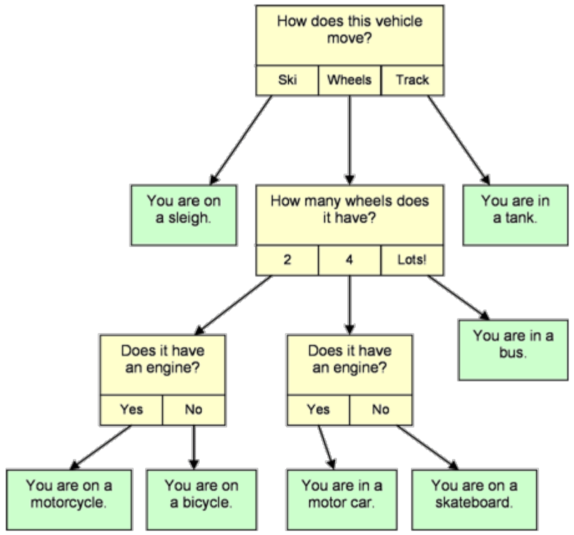
What is the current state of the art?



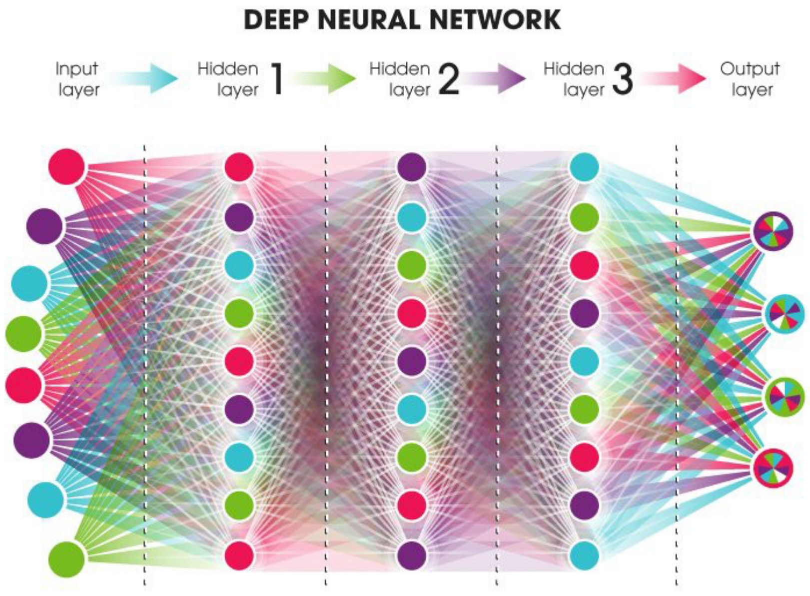
Black-box statistical predictions are inadequate

Explanations must be understandable to non-specialist

Tradeoff:



- OR -



neuralnetworksanddeeplearning.com - Michael Nielsen, Yoshua Bengio, Ian Goodfellow, and Aaron Courville, 2016.

Expert system:
Good for explanations,
not so good for accuracy

Neural nets:
Good for accuracy,
not so good for explanations

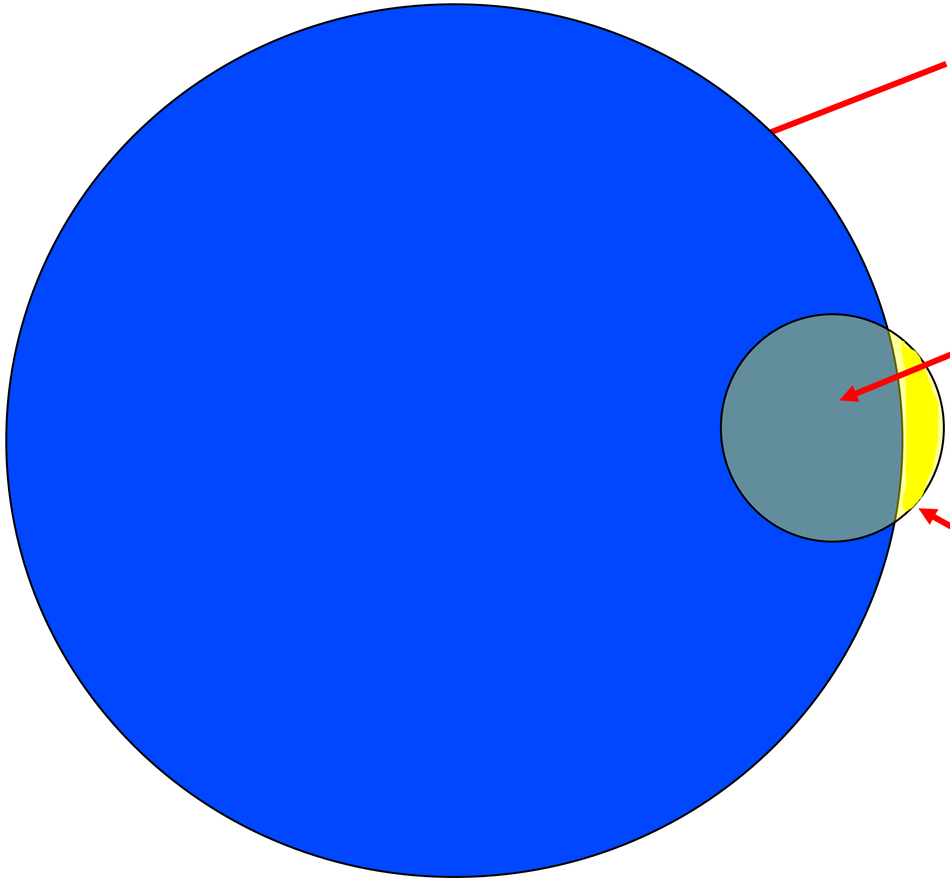
How do we get the best of both worlds?

What can NIST do?

- The classification problem in machine learning is closely related to the problem of fault location in combinatorial testing for software.
- The objective in both cases is to identify a small number of interactions, out of possibly billions or more, that trigger a failure (in combinatorial testing) or produce a conclusion (in machine learning).
- We have methods and tools for fault location in combinatorial testing that could be adapted to ML problems, to identify the rare combinations of variable values that produce conclusions in AI systems.
- This approach has not been applied to AI/ML before.
- NIST has established the leading project in combinatorial software testing

Fault location

Given: a set of tests that the SUT fails, which combinations of variables/values triggered the failure?

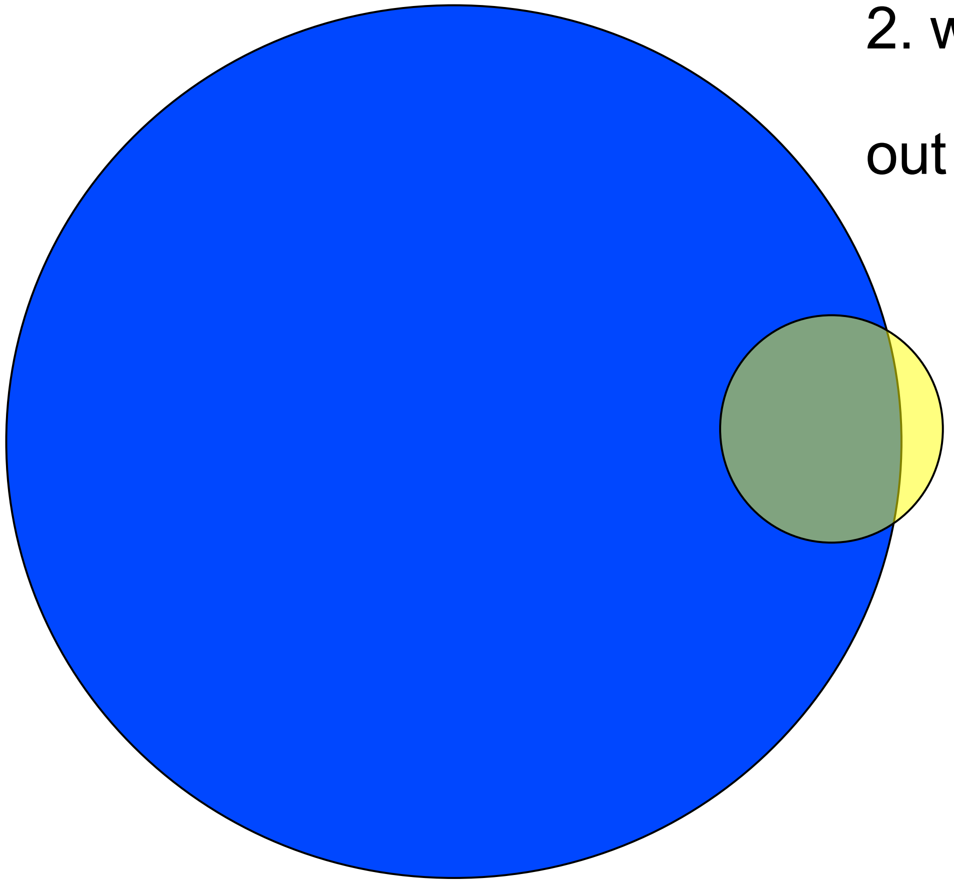


variable/value combinations in passing tests

variable/value combinations in failing tests

Combinations in failing but not in passing tests
These are the ones we want – how do we find them?

Fault location – what's the problem?



If they're in failing set but not in passing set:

1. which ones triggered the failure?
2. which ones don't matter?

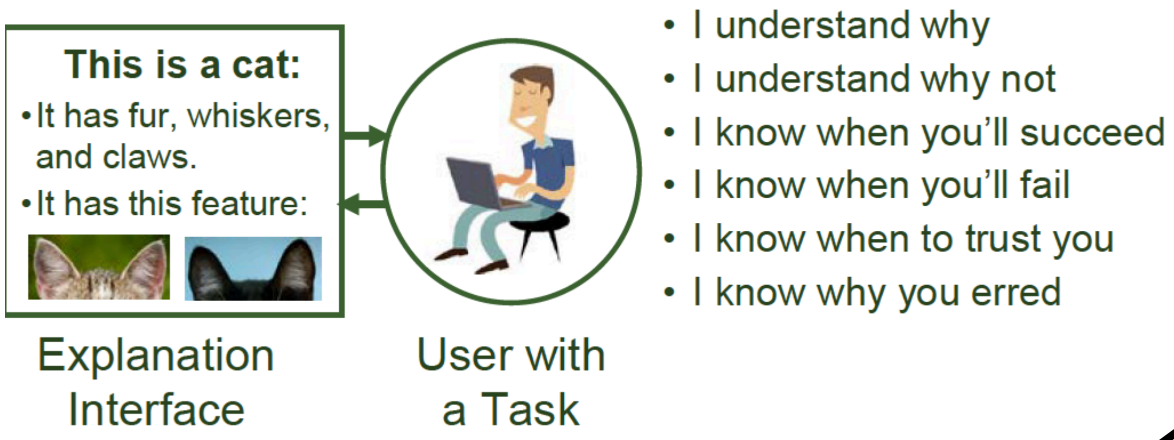
out of $v^t \binom{n}{t}$ combinations

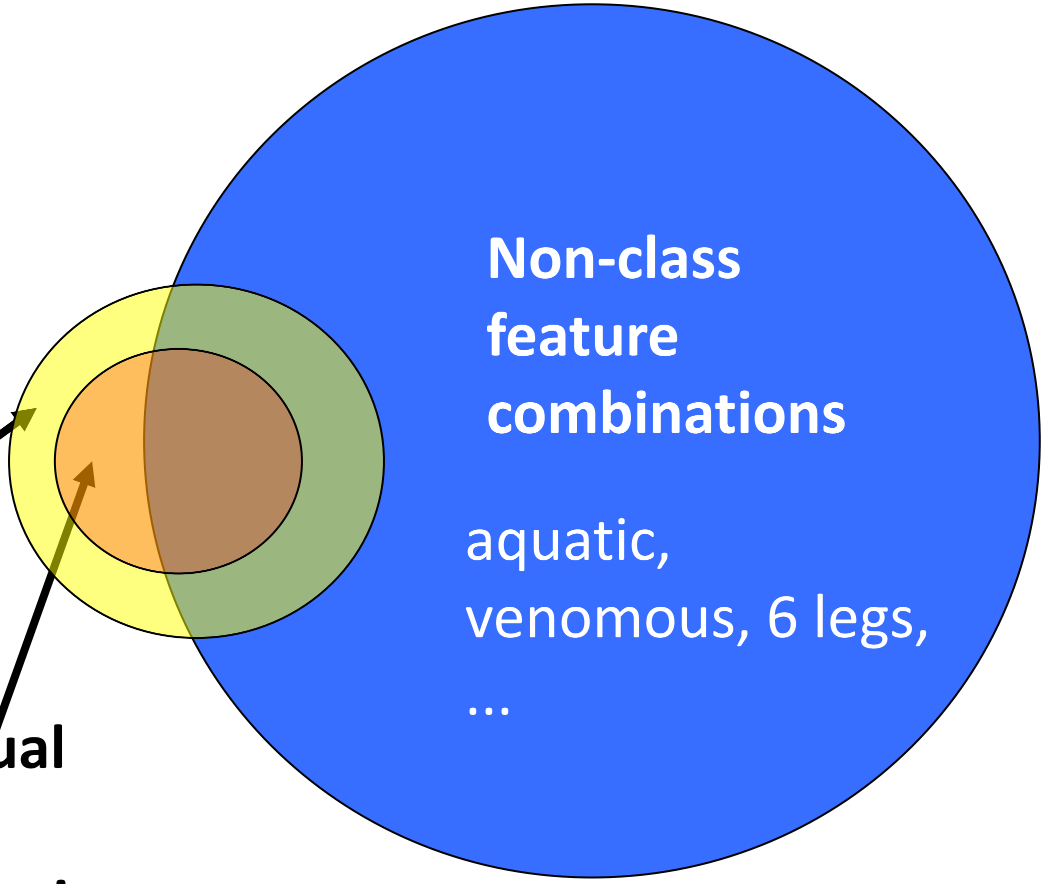
Example:
30 variables, 5 values each,
input configuration 5^{30}
➔ 445,331,250 5-way combinations

142,506 combinations in <u>each test</u>

FInd one or two out of >142,000 that caused failure

Relevance to explainable AI





Class feature combinations - brown & furry, black & furry, whiskers, claws, **…** not aquatic, not venomous, not 6 legs,

Individual feature combinations – brown & furry, whiskers, claws, not aquatic, not venomous, not 6 legs, **…**

Animal shares features with <u>cat</u> class

Animal does not share features with <u>non-cat</u> classes

Why is this creature recognized as a reptile?

Class File: Class file rep1.csv; rows=1; cols=16

Nominal File: Nominal file notreptile.csv; rows=96; cols=16 || 2-way: 120 3-way: 560 4-way: 1,820 5-way: 4,368 6-way: 8,008

Class File Contents:

hair	feathers	eggs	milk	airborne	aquatic	predator	toothed	backbone	breathes	venomous	fins	nlegs	tail	domestic	catsize
0	0	1	0	0	0	0	0	1	1	0	0	4	1	0	1

Input configuration $2^{15}6^{1}$

```
---------------
0053 occurrences = 0.552 of cases, hair = 0
0076 occurrences = 0.792 of cases, feathers = 0
0055 occurrences = 0.573 of cases, eggs = 1
0055 occurrences = 0.573 of cases, milk = 0
0072 occurrences = 0.750 of cases, airborne = 0
0061 occurrences = 0.635 of cases, aquatic = 0
0044 occurrences = 0.458 of cases, predator = 0
0039 occurrences = 0.406 of cases, toothed = 0
0078 occurrences = 0.813 of cases, backbone = 1
0076 occurrences = 0.792 of cases, breathes = 1
0090 occurrences = 0.938 of cases, venomous = 0
0079 occurrences = 0.823 of cases, fins = 0
0036 occurrences = 0.375 of cases, nlegs = 4
0070 occurrences = 0.729 of cases, tail = 1
0083 occurrences = 0.865 of cases, domestic = 0
0043 occurrences = 0.448 of cases, catsize = 1
```

No single feature is sufficient explanation – shares features with non-reptiles

No pair of features sufficient – shares 2-way combinations w/ non-reptiles

```
0002 occurrences = 0.021 of cases, toothed,nlegs = 0,4
0005 occurrences = 0.052 of cases, hair,nlegs = 0,4
0005 occurrences = 0.052 of cases, milk,nlegs = 0,4
0006 occurrences = 0.063 of cases, eggs,nlegs = 1,4
0008 occurrences = 0.083 of cases, toothed,catsize = 0,1
0011 occurrences = 0.115 of cases, milk,catsize = 0,1
0012 occurrences = 0.125 of cases, eggs,catsize = 1,1
0013 occurrences = 0.135 of cases, hair,catsize = 0,1
0015 occurrences = 0.156 of cases, predator,catsize = 0,1
```

3-way combinations produce rules to explain recognition of Testudo as a reptile

```
00000 occurrences = 0.000 of cases, aquatic,toothed,nlegs = 0,0,4
00000 occurrences = 0.000 of cases, eggs,aquatic,nlegs = 1,0,4
00000 occurrences = 0.000 of cases, hair,aquatic,nlegs = 0,0,4
00000 occurrences = 0.000 of cases, hair,nlegs,catsize = 0,4,1
00000 occurrences = 0.000 of cases, milk,aquatic,nlegs = 0,0,4
00000 occurrences = 0.000 of cases, milk,nlegs,catsize = 0,4,1
00000 occurrences = 0.000 of cases, predator,toothed,nlegs = 0,0,4
00001 occurrences = 0.010 of cases, eggs,nlegs,catsize = 1,4,1
00001 occurrences = 0.010 of cases, eggs,predator,nlegs = 1,0,4
00001 occurrences = 0.010 of cases, feathers,toothed,backbone = 0,0,1
```

Non-reptiles in the database do not have these 3-way combinations

Only reptiles have these combinations of features:

not aquatic AND not toothed AND four legs

egg-laying AND not aquatic AND four legs

not hairy AND four legs AND cat size

not milk-producing AND not aquatic AND four legs

not milk-producing AND four legs AND cat size

not predator AND not toothed AND four legs

Sample ML problem

- “Titanic survivors” – popular demo problem for ML
- Predict which passengers survive, using attributes:
- Passenger class: 1^{st}, 2^{nd}, 3rd
- Sex
- Age: 14 and under, 15 to 20, 21 to 70, over 70
- Number of siblings or spouses onboard
- Number of parents or children onboard
- Embarkation point: Southampton, England; Queenstown, Ireland; Cherbourg, France
- Input configuration $2^1 3^2 4^1 5^2$

Example using prototype – what factors explain this passenger's survival?

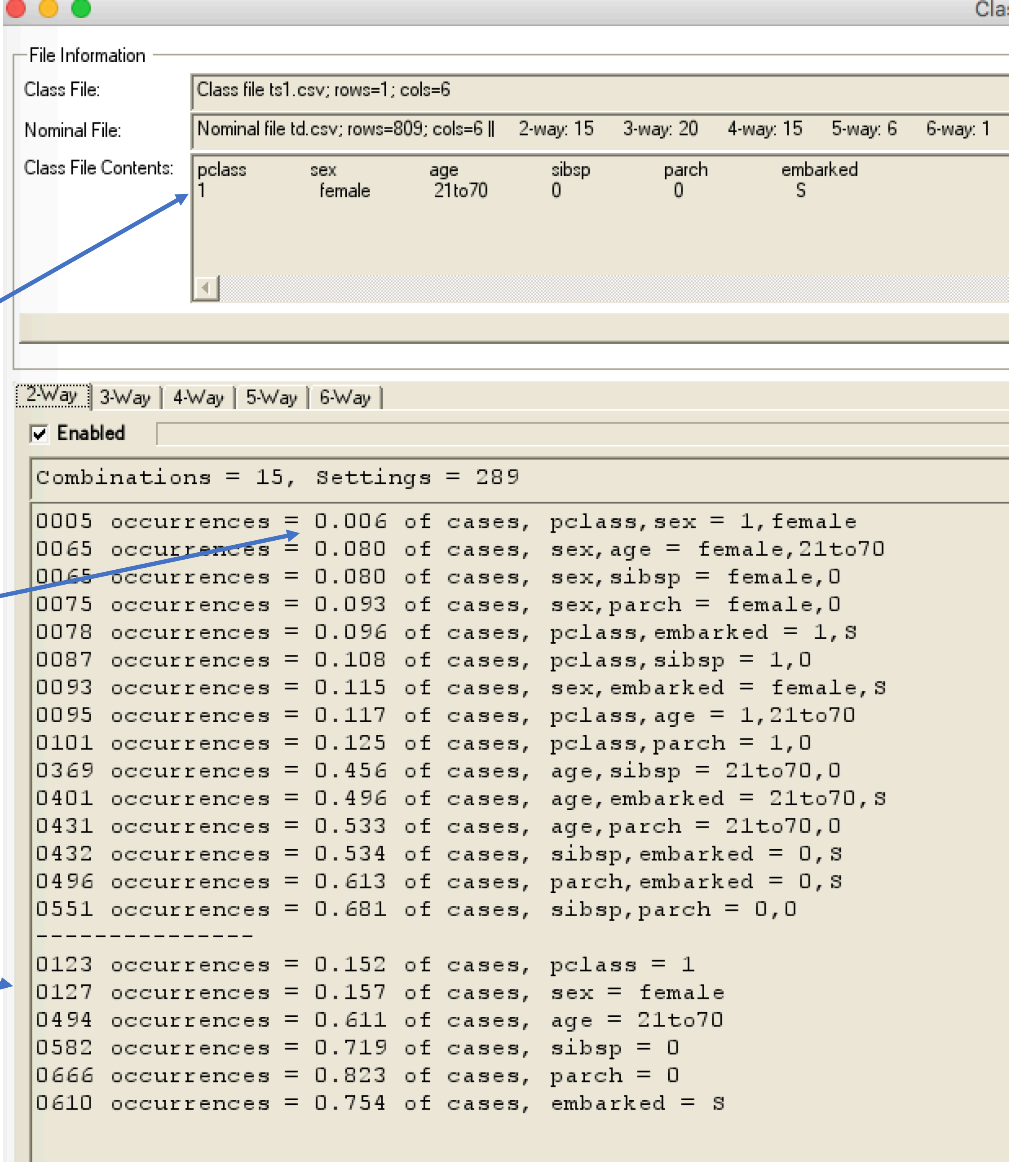
First class passenger, female aged 21 to 70, no siblings, spouse, parents, or children onboard, from England

What factors differentiate passenger from casualties?

Consider 2-way combinations of factors:
1st class female passengers (like this one) were only 0.6% of casualties

No single factor is adequate explanation:
15% of dead were 1st class;
16% were female;
61% aged 21 to 70

Survival explained by being female passenger traveling first class. Neither of these two factors alone is enough.



Heatmap visualization of factor combinations

Psngr class	Sex	Age	# sibling spouse	#parent child	embarked
1	f	21to70	0	0	S

Green to red -> **more significant** to **less significant** for explanation

Heatmap	female	21to070	no sibling/spouse	no parents/children	Southampton
1st class	0.0062	0.1174	0.1075	0.1248	0.0964
female		0.0803	0.0803	0.0927	0.1150
21 to 70			0.4561	0.5328	0.4957
no sibling/ spouse				0.6811	0.5340
no parents/ children					0.6131

Another example– what factors explain this passenger's survival?

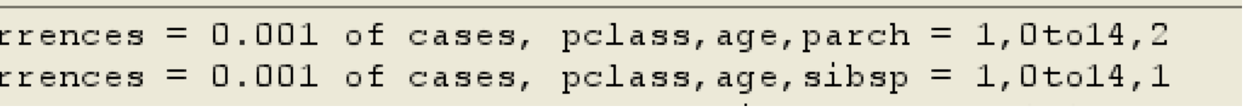
First class passenger, male child, with one sibling, two parents onboard, from England

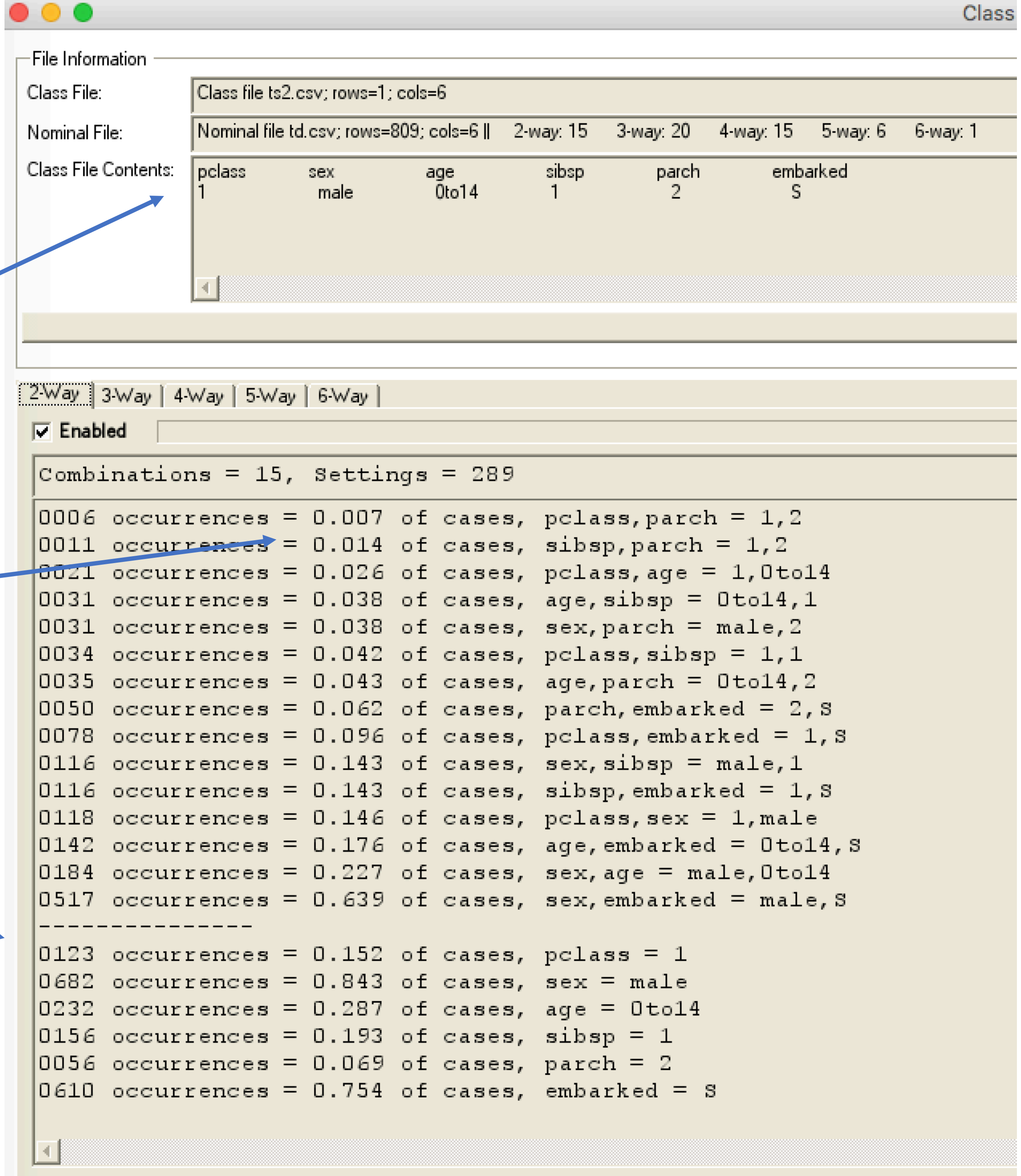
What factors differentiate passenger from casualties?

Consider 2-way combinations of factors:
1st class passengers with two parents onboard (like this one) were only 0.7% of casualties

No single factor is adequate explanation:
15% of dead were 1st class;
84% were male;
29% were children 14 and under

Survival explained by being child with parents traveling first class. <u>No single factor alone is enough.</u>
<u>Easily seen in 3-way combinations:</u>





Mapping combinations to expressions

- Report identifies t-way combinations that distinguish the predicted class from others
- Combinations can be mapped to expressions to produce a rule-based type of explanation

if (1^{st} class passenger AND female) OR (female AND age 21to70) OR (female AND no siblings/spouses) then SURVIVE

if (1^{st} class passenger AND age 14 or under AND parents onboard) OR (1^{st} class passenger AND age 14 or under AND siblings onboard) then SURVIVE

As noted, none of the single factors above is sufficient for explanation

Example: empty vs. occupied rooms, using sensor data

Class File: Class file o1.csv; rows=1; cols=5

Nominal File: Nominal file empty.csv; rows=7703; cols=5 || 2-way: 10 3-way: 10 4-way: 5 5-way: 1 6-way: 0

Class File Contents:

Temperature	Humidity	Light	CO2	HumidityRatio
B3	B3	B2	B2	B4

2-Way | 3-Way | 4-Way | 5-Way | 6-Way

☑ Enabled

```
Combinations = 10, Settings = 210

0016 occurrences = 0.002 of cases, Humidity,Light = B3,B2
0016 occurrences = 0.002 of cases, Light,CO2 = B2,B2
0036 occurrences = 0.005 of cases, Temperature,Light = B3,B2
0040 occurrences = 0.005 of cases, CO2,HumidityRatio = B2,B4
0043 occurrences = 0.006 of cases, Light,HumidityRatio = B2,B4
0054 occurrences = 0.007 of cases, Temperature,CO2 = B3,B2
0078 occurrences = 0.010 of cases, Humidity,CO2 = B3,B2
0205 occurrences = 0.027 of cases, Temperature,HumidityRatio = B3,B4
0247 occurrences = 0.032 of cases, Temperature,Humidity = B3,B3
0495 occurrences = 0.064 of cases, Humidity,HumidityRatio = B3,B4
---------------
0523 occurrences = 0.068 of cases, Temperature = B3
2415 occurrences = 0.314 of cases, Humidity = B3
0085 occurrences = 0.011 of cases, Light = B2
0534 occurrences = 0.069 of cases, CO2 = B2
2190 occurrences = 0.284 of cases, HumidityRatio = B4
```

Why do we conclude this room is occupied?

These levels of humidity and lighting are strong indication

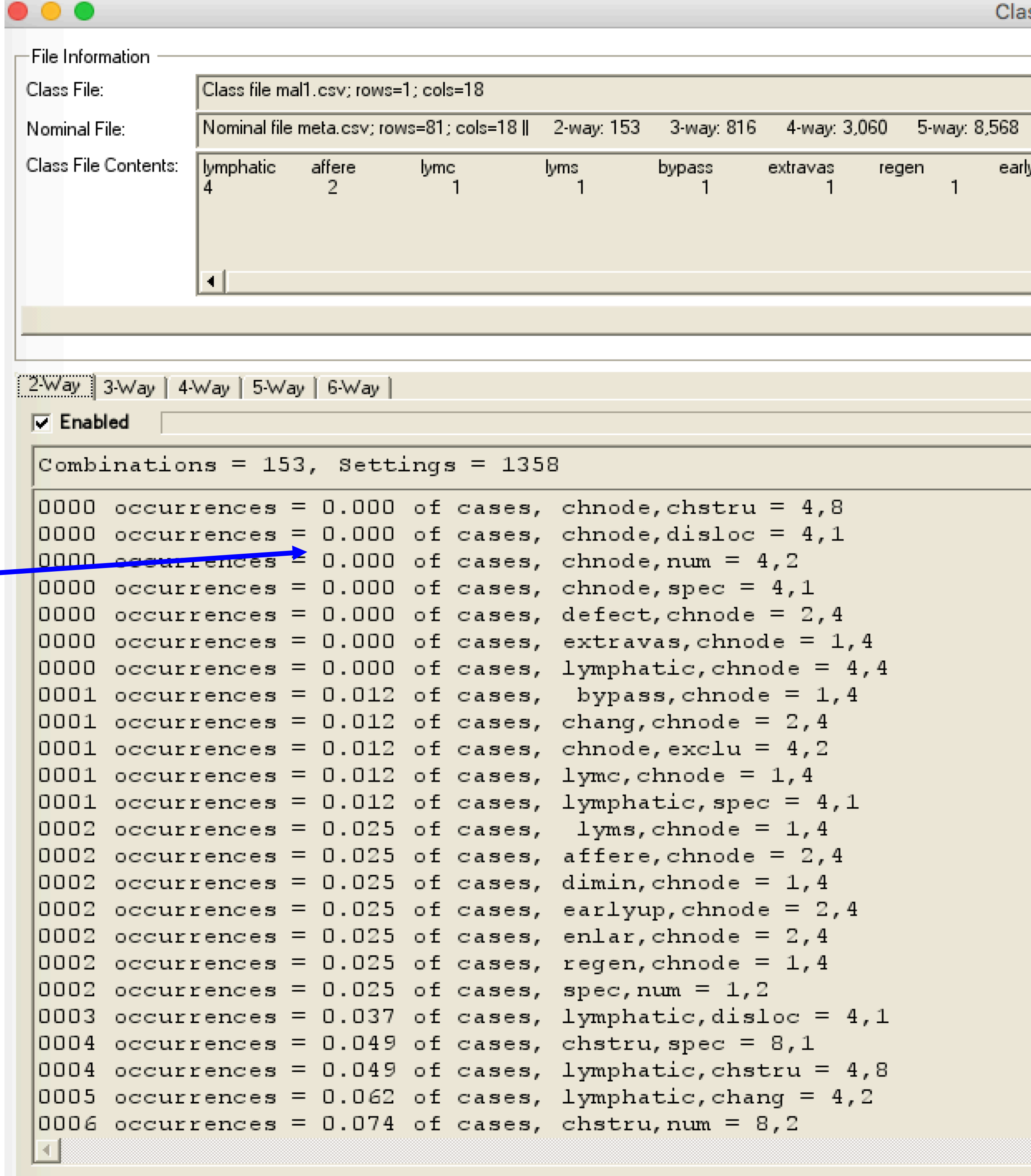
Considering levels of lighting, CO2, and humidity ratio provide even stronger evidence:

<u>Empty rooms don't have these levels</u>

```
00003 occurrences = 0.000 of cases, Light,CO2,HumidityRatio = B2,B2,B4
00005 occurrences = 0.001 of cases, Humidity,Light,CO2 = B3,B2,B2
00008 occurrences = 0.001 of cases, Temperature,Light,CO2 = B3,B2,B2
00011 occurrences = 0.001 of cases, Humidity,Light,HumidityRatio = B3,B2,B4
```

A different example: lymph node pathology – why is this classified as malignant not metastatic?

- These combinations are characteristic of lymphoma that arises in lymph node instead of metastatic that spread to node from somewhere else



Summary

- Combinatorial methods can provide explainable AI

- We have prototype that applies this approach
  - Determine combinations of variable values that differentiate an example from other possible conclusions
    - ➔ Feature combinations present shared with class
    - ➔ Feature combinations not shared with class not present

- Method can be applied to black-box functions such as CNNs

- Present explanation in the preferred form of rules,
  "if A & B, or C with D & E, then conclusion is X"

Summary

- Explainability is a critical problem in the acceptance of artificial intelligence/machine learning, especially for critical applications
- Human users will not trust AI if conclusions cannot be explained
- Methods from combinatorial software testing can be applied to solving the problem of explainable AI
- We have prototype that applies this approach
  - Determine combinations of variable values that differentiate an example from other possible conclusions
    - ➔ Feature combinations present shared with class
    - ➔ Feature combinations not shared with class not present
  - Present explanation in the preferred form of rules, “if A & B, or C with D & E, then conclusion is X”
- Method can be applied to black-box functions such as CNNs

What has been tried?

- Interpretable models – e.g. rule-based expert systems: "if patient has symptoms A and B, or has B with C and D, then illness is X"
  - best for explanations
  - hard to find rules
  - less accurate than other approaches
- Modify neural nets etc. to add explanations
  - reduces accuracy, complicates the system
  - explanations still not very understandable
- Model induction - infer explainable model from black-box
  - flexible for application, good explanations using only input, output
  - hard to produce the explainable model
- Our approach – derive rule predicates from inputs and outputs to CNNs and other black-box functions