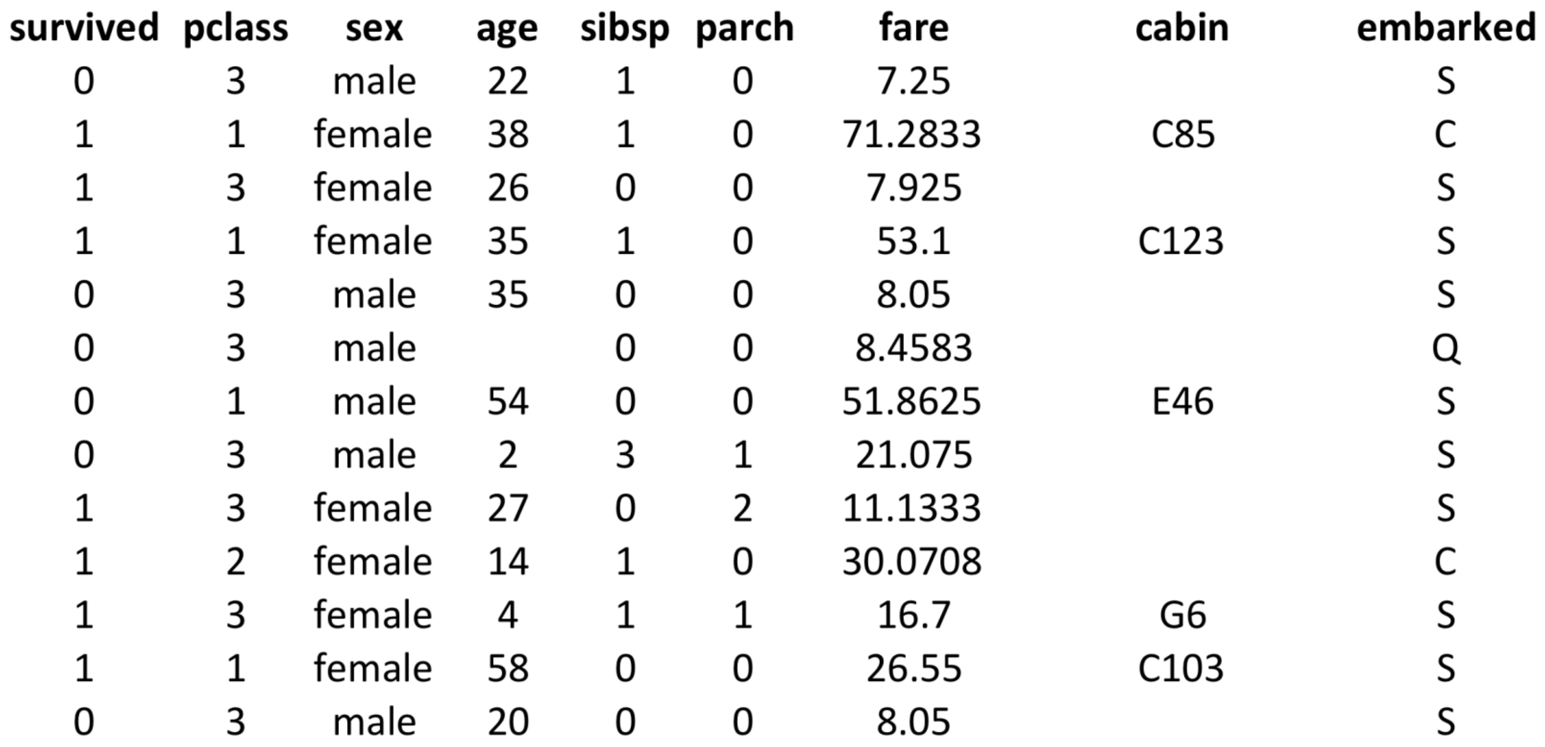
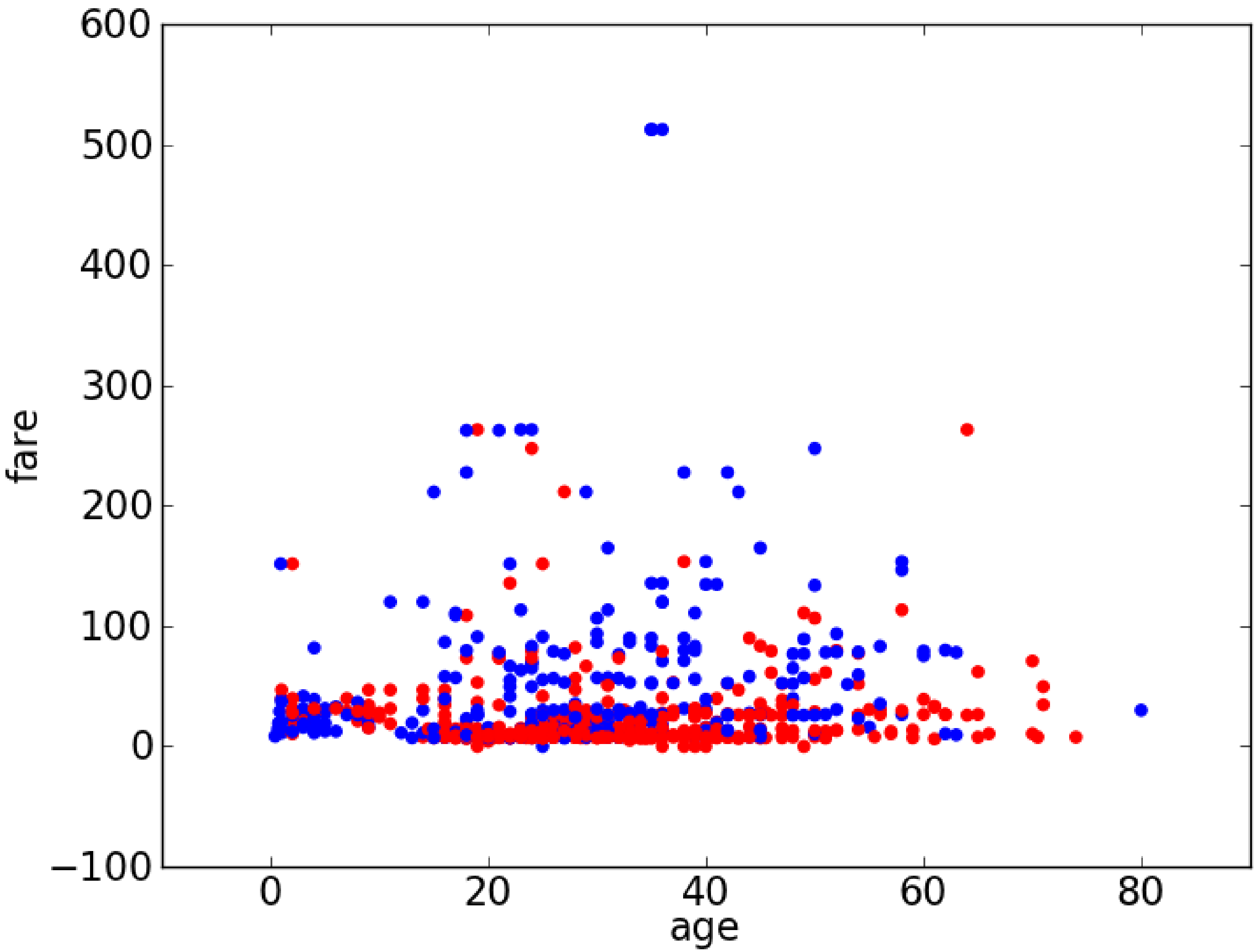
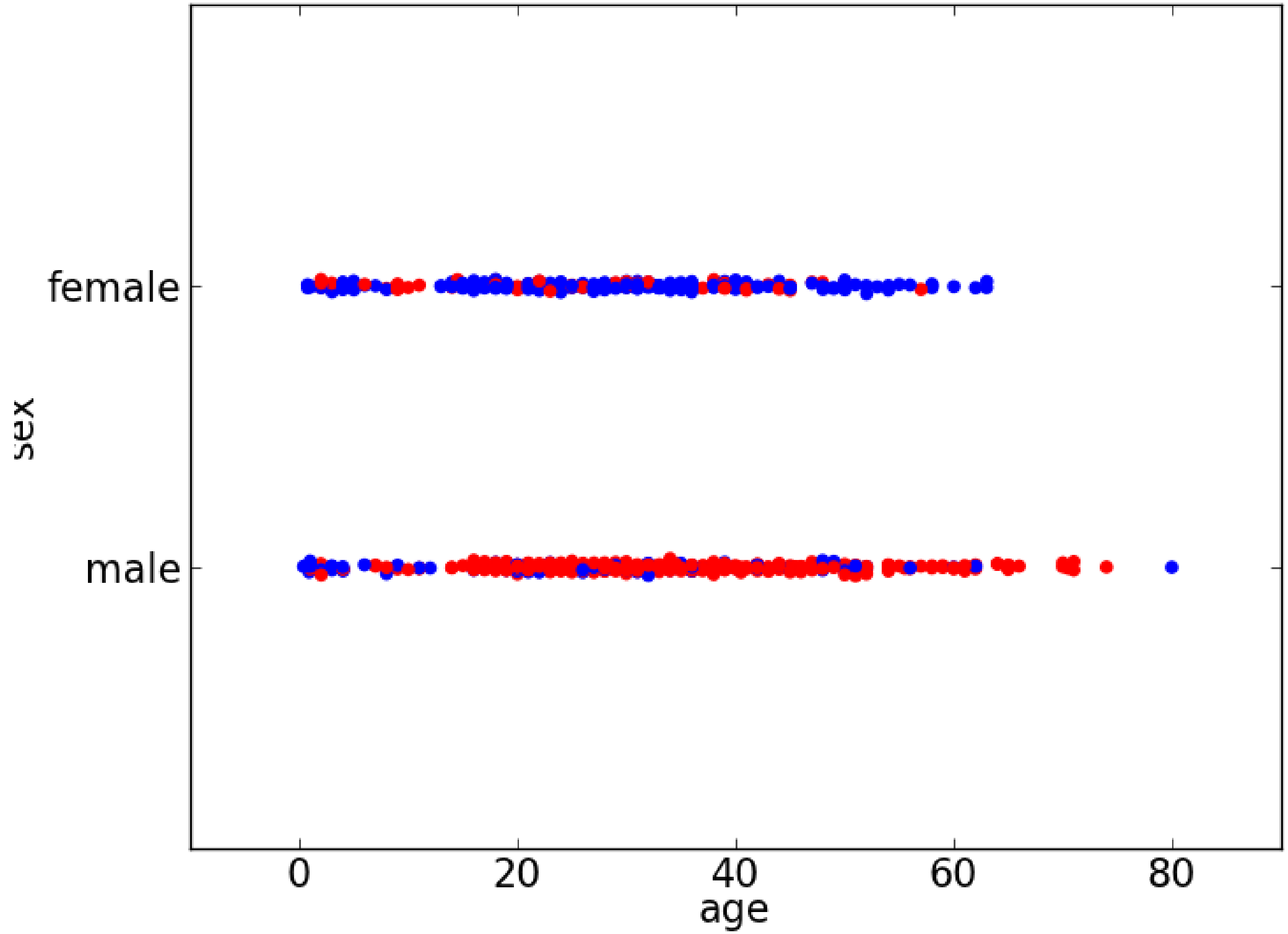
Titanic Dataset

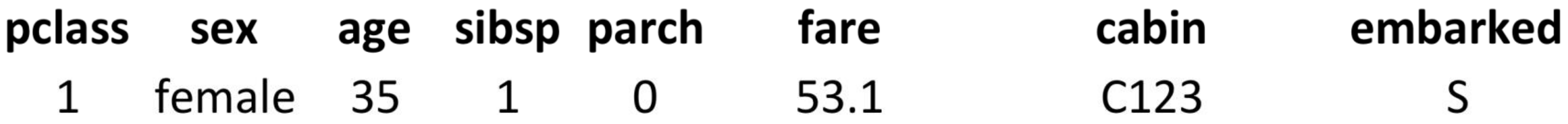


survived	pclass	sex	age	sibsp	parch	fare	cabin	embarked
0	3	male	22	1	0	7.25		S
1	1	female	38	1	0	71.2833	C85	C
1	3	female	26	0	0	7.925		S
1	1	female	35	1	0	53.1	C123	S
0	3	male	35	0	0	8.05		S
0	3	male		0	0	8.4583		Q
0	1	male	54	0	0	51.8625	E46	S
0	3	male	2	3	1	21.075		S
1	3	female	27	0	2	11.1333		S
1	2	female	14	1	0	30.0708		C
1	3	female	4	1	1	16.7	G6	S
1	1	female	58	0	0	26.55	C103	S
0	3	male	20	0	0	8.05		S


600
500
400
300
200
100
0
−100
fare
0
20
40
60
80
age


female
male
sex
0
20
40
60
80
age

A very naïve classifier



pclass	sex	age	sibsp	parch	fare	cabin	embarked
1	female	35	1	0	53.1	C123	S

Does the new data point x^* ***exactly*** match a previous point x_i?

If so, assign it to the same class as x_i

Otherwise, just guess.

This is the “rote” classifier

A minor improvement

pclass	sex	age	sibsp	parch	fare	cabin	embarked
1	female	35	1	0	53.1	C123	S

Does the new data point x^* match a set pf previous points x_i on some specific attribute?

If so, take a vote to determine class.

Example: If most females survived, then assume every female survives

But there are lots of possible rules like this.
And an attribute can have more than two values.

If most people under 4 years old survive, then assume everyone under 4 survives
If most people with 1 sibling survive, then assume everyone with 1 sibling survives

How do we choose?

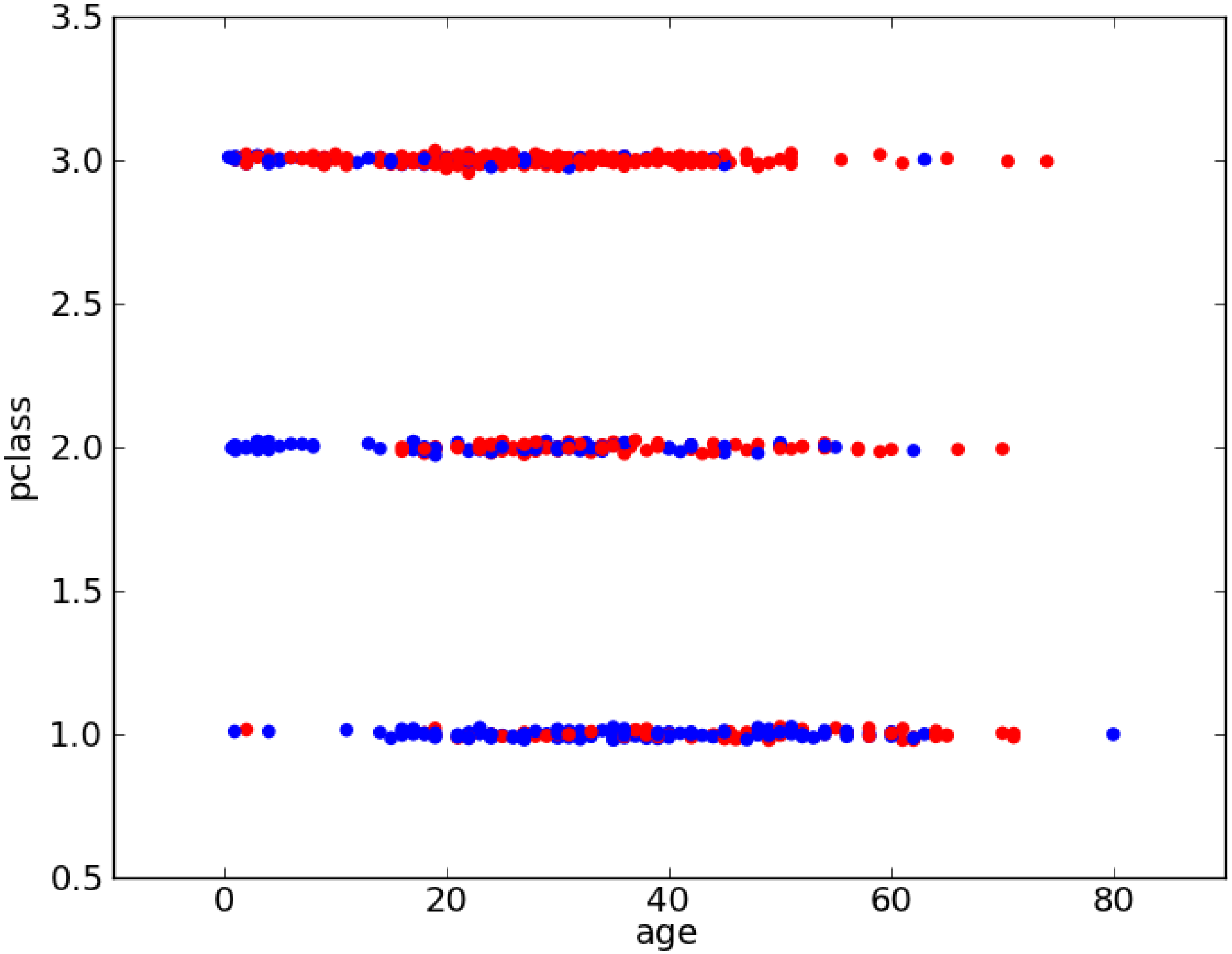
IF sex=‘female’ THEN survive=yes
ELSE IF sex=‘male’ THEN survive = no

```
confusion matrix

no   yes    <-- classified as
468  109  |   no
 81  233  |  yes
```

(468 + 233) / (468+109+81+233) = 79% correct (and 21% incorrect)

Not bad!


3.5
3.0
2.5
2.0
1.5
1.0
0.5
pclass
0
20
40
60
80
age

```
IF pclass=‘1’ THEN survive=yes
ELSE IF pclass=‘2’ THEN survive=yes
ELSE IF pclass=‘3’ THEN survive=no
```

```
confusion matrix

no   yes    <-- classified as
372  119  |   no
177  223  |  yes
```

(372 + 223) / (372+119+223+177) = 67% correct (and 33% incorrect)

a little worse

1-Rule

```
For each attribute A:
     For each value V of that attribute, create a rule:
       1. count how often each class appears
       2. find the most frequent class, c
       3. make a rule "if A=V then Class=c"
     Calculate the error rate of this rule

   Pick the attribute whose rules produce the lowest error rate
```