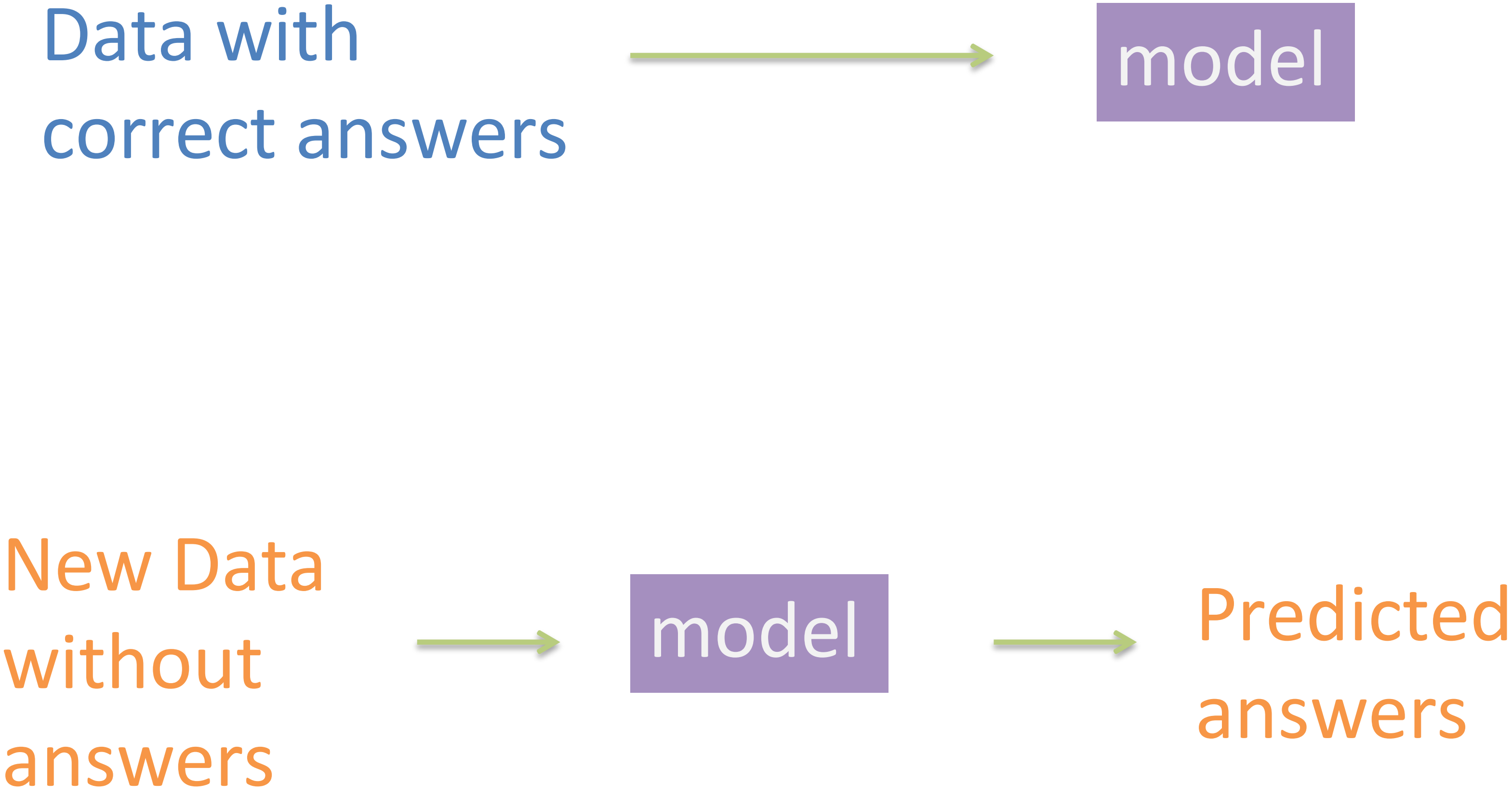
Supervised Learning

DATA SCIENCE BOOTCAMP

What is Supervised Learning?



Data with correct answers

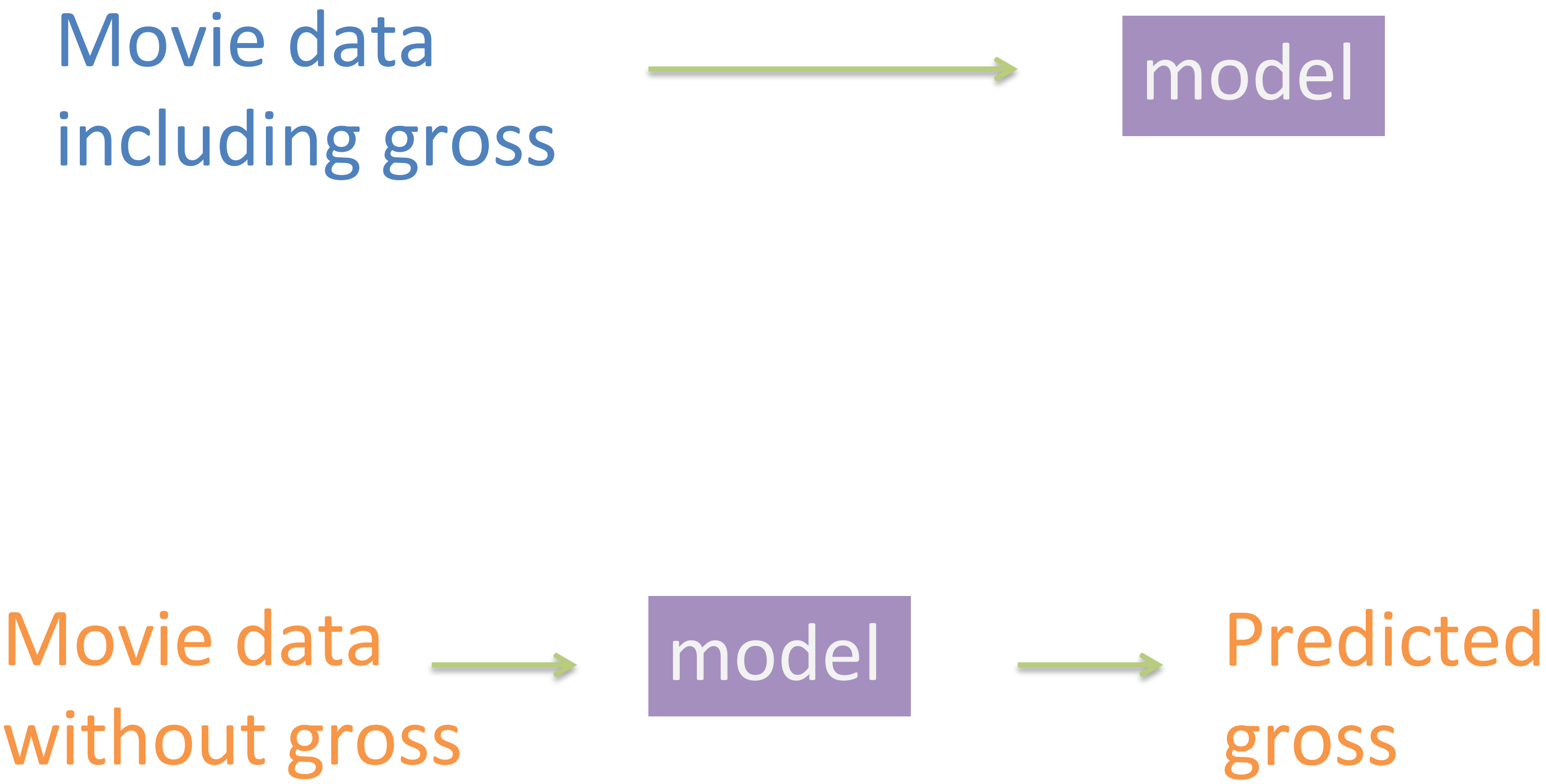
model

New Data without answers

model

Predicted answers

Regression: “answers” are numeric



Movie data including gross

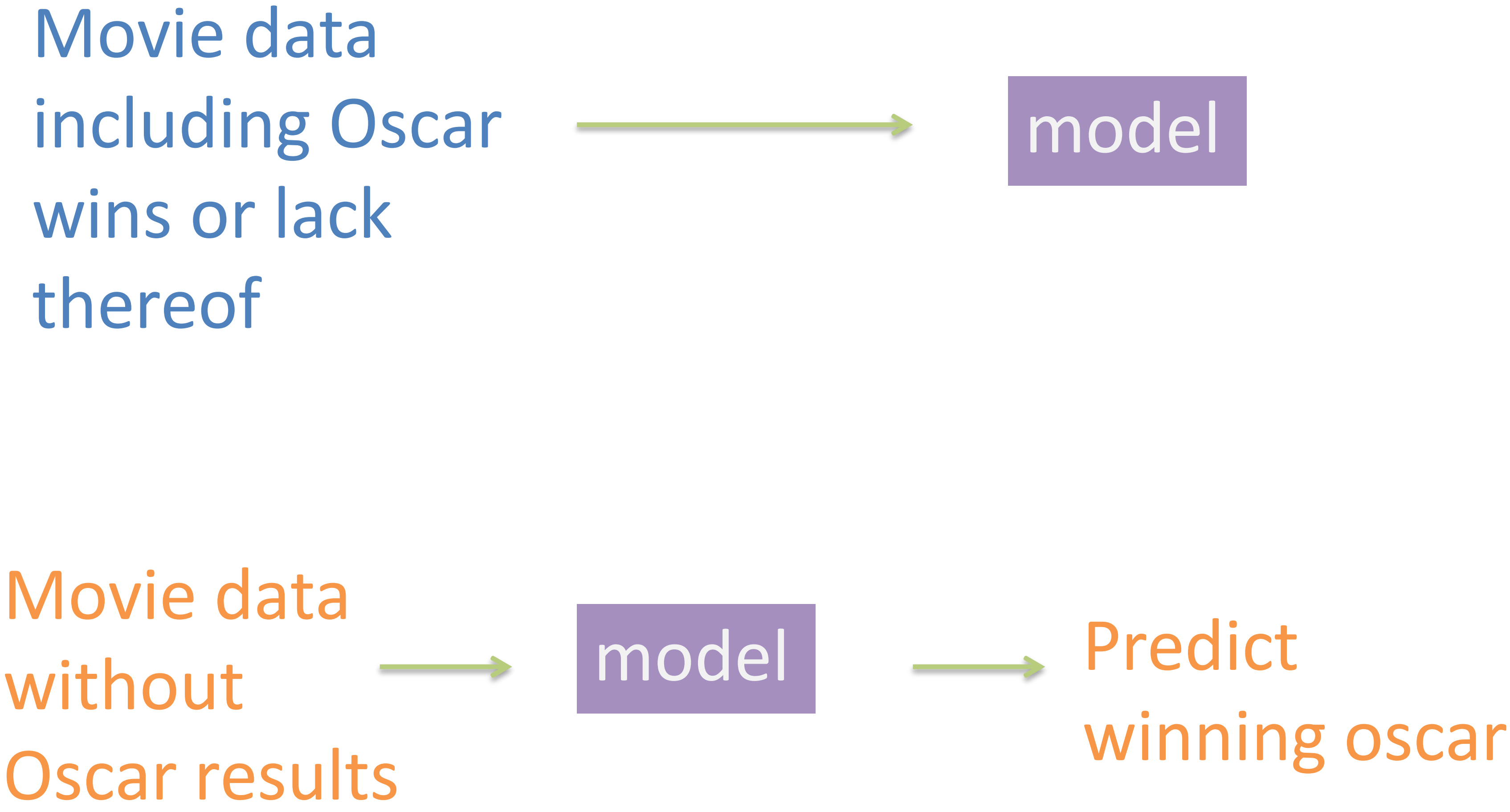
model

Movie data without gross

model

Predicted gross

Classification: “answers” are categories



Movie data including Oscar wins or lack thereof

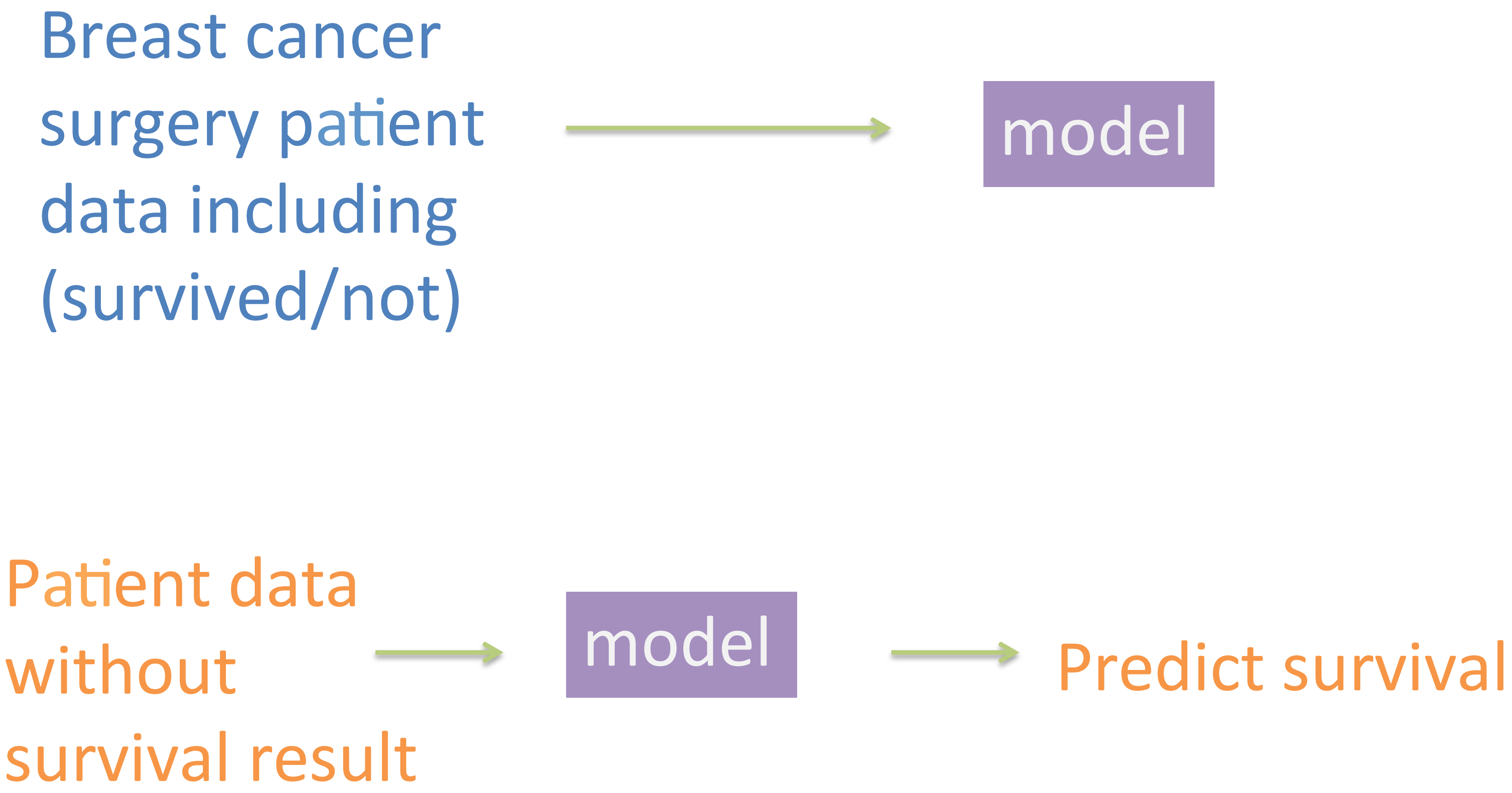
model

Movie data without Oscar results

model

Predict winning oscar

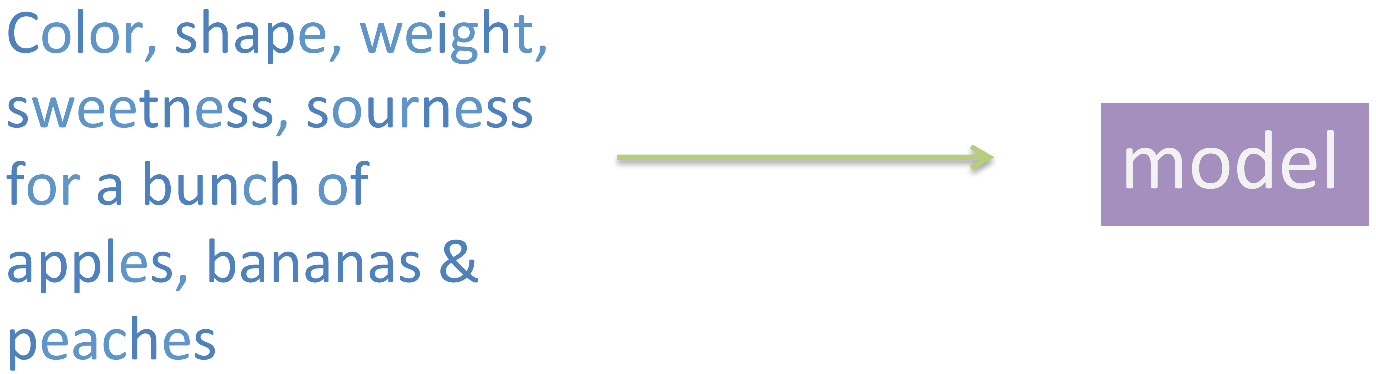
Classification: “answers” are categories



Breast cancer surgery patient data including (survived/not) → model

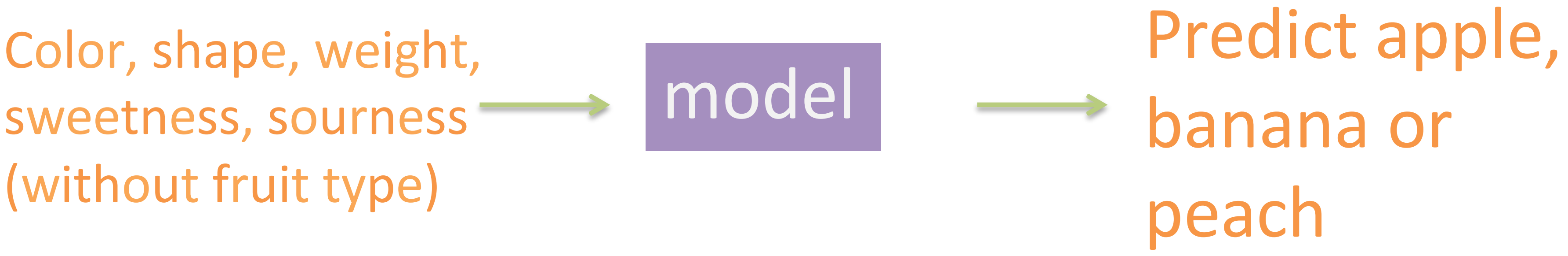
Patient data without survival result → model → Predict survival

Classification: "answers" are categories



Color, shape, weight, sweetness, sourness for a bunch of apples, bananas & peaches

model

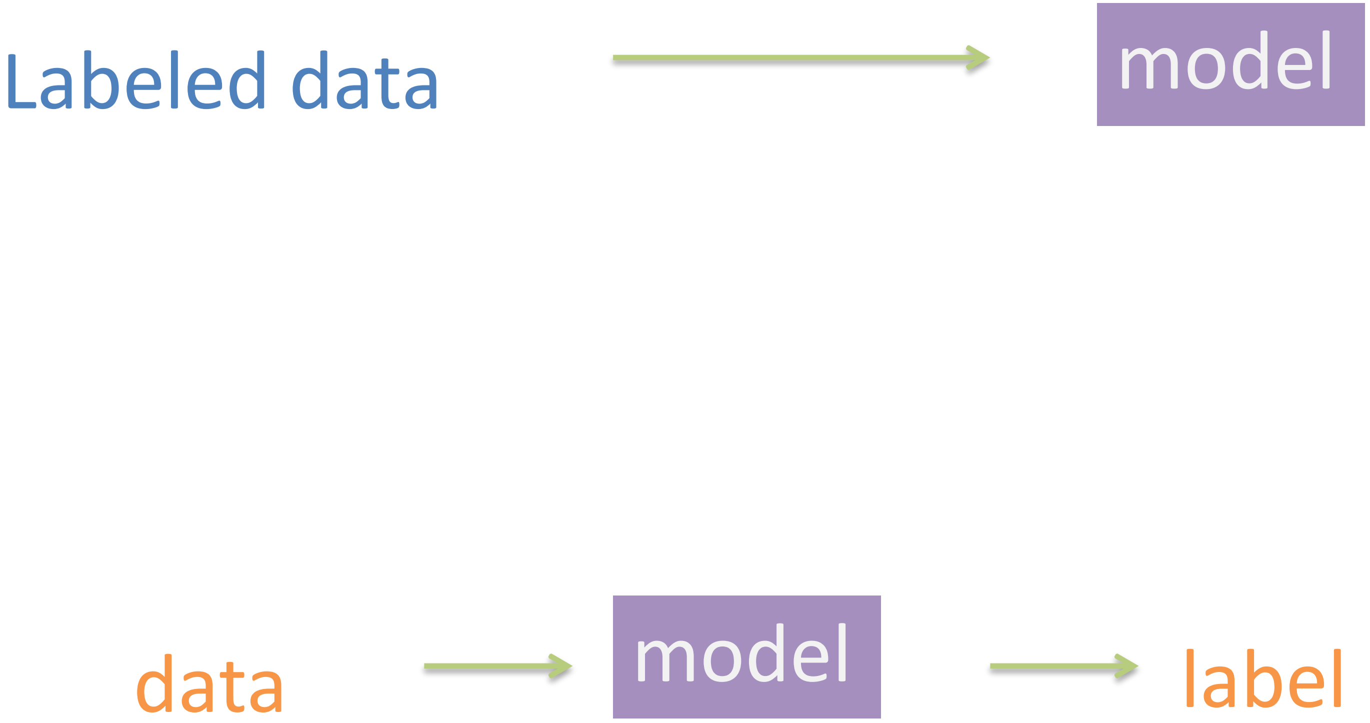


Color, shape, weight, sweetness, sourness (without fruit type)

model

Predict apple, banana or peach

Classification: “answers” are categories



Labeled data → model

data → model → label

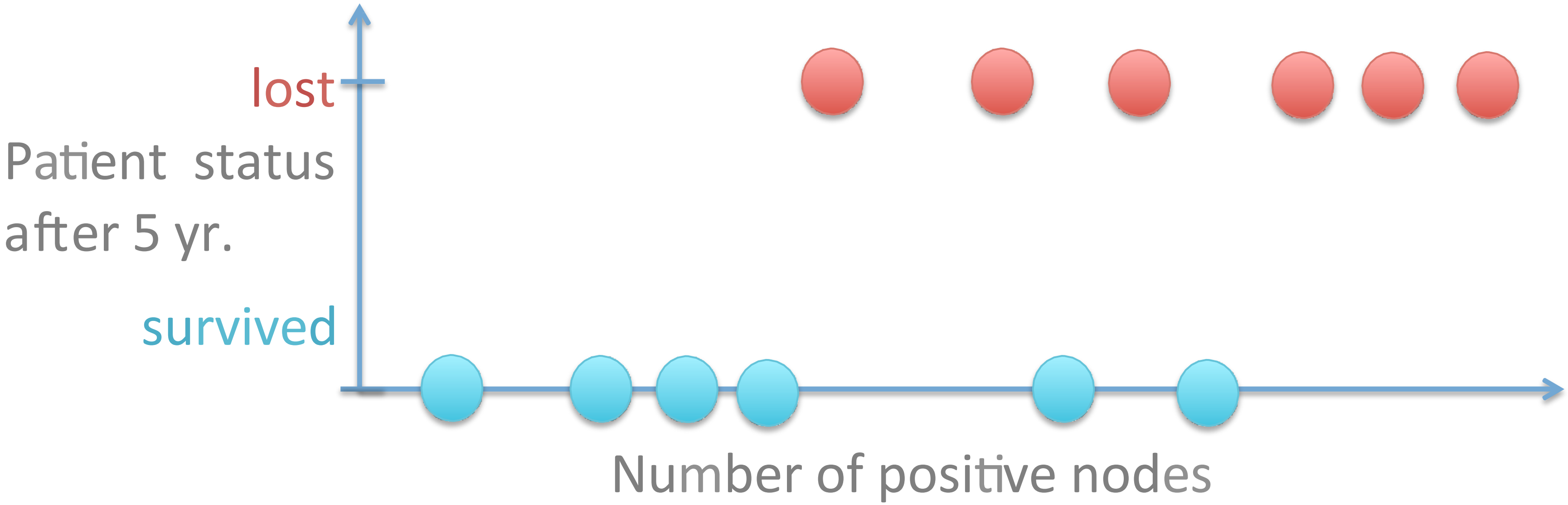
Example: each data point (row)

Label: each category to be predicted

Feature: each property (column) used in predicting

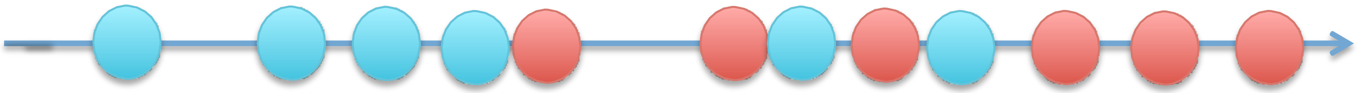
1 Feature: Number of + nodes

2 Labels: Survived / Lost



1 Feature: Number of + nodes

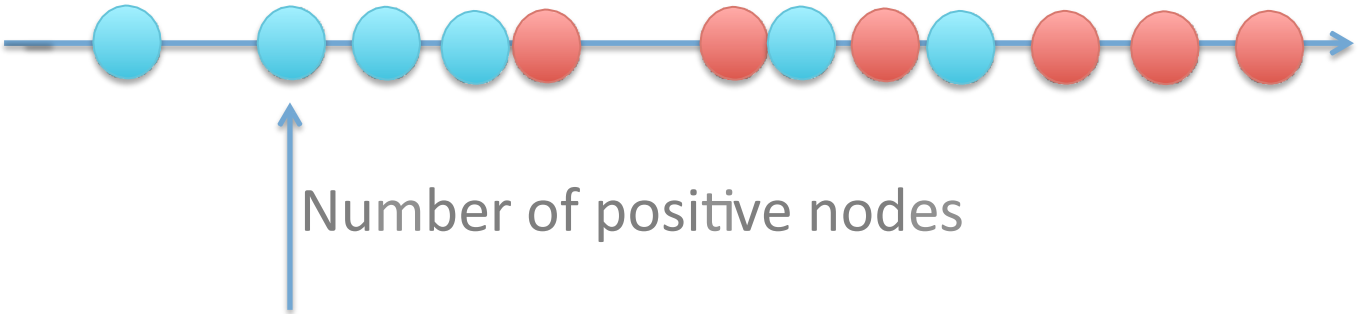
2 Labels: Survived / Lost



Number of positive nodes

1 Feature: Number of + nodes

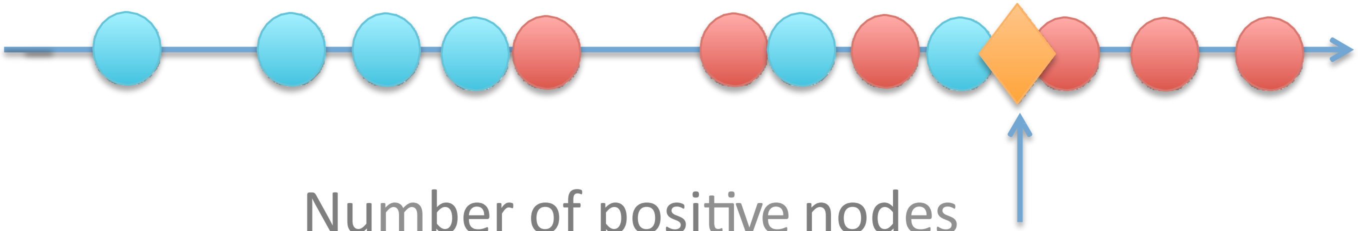
2 Labels: Survived / Lost



an example

1 Feature: Number of + nodes

2 Labels: Survived / Lost

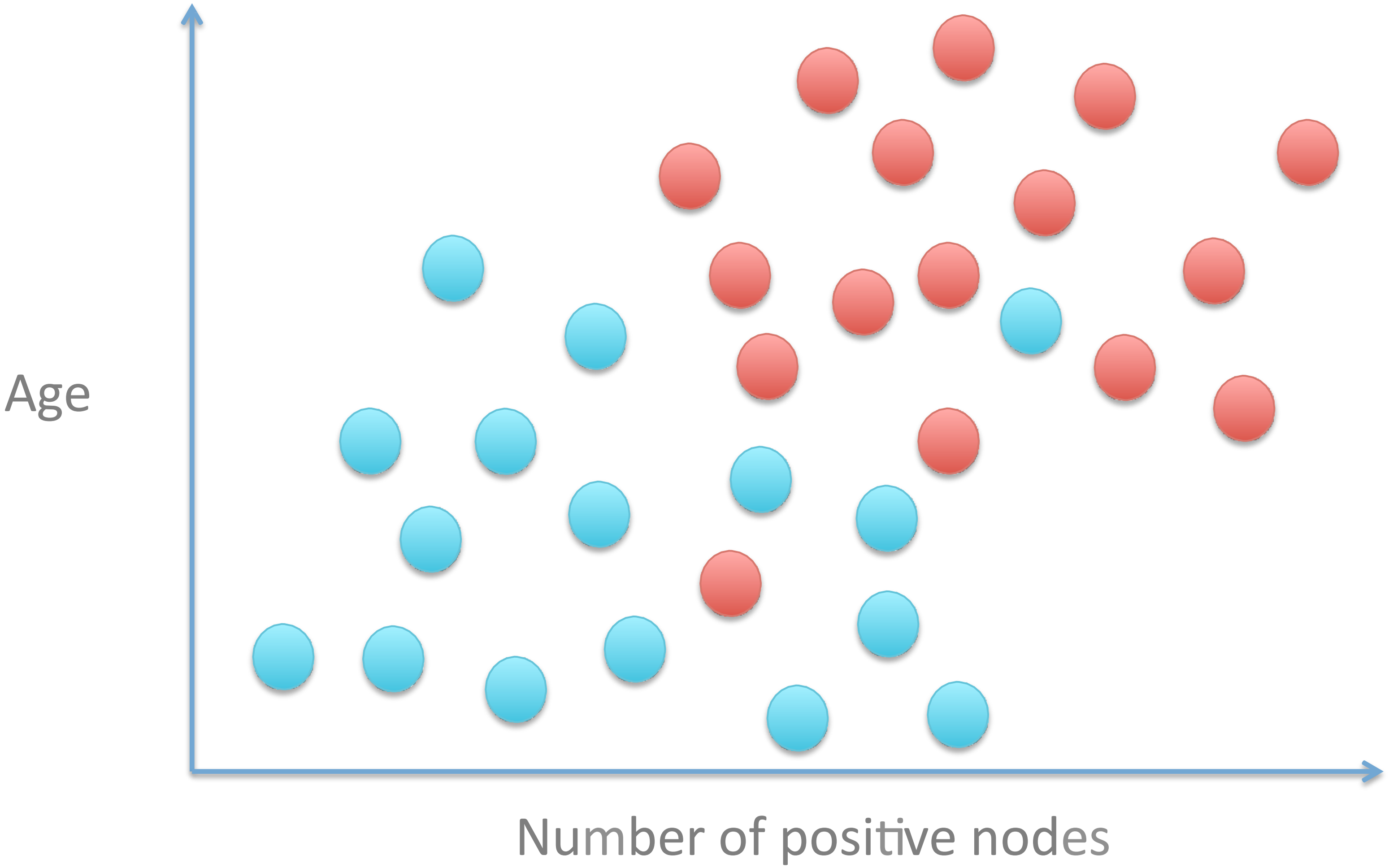


Number of positive nodes

new example
(predict)

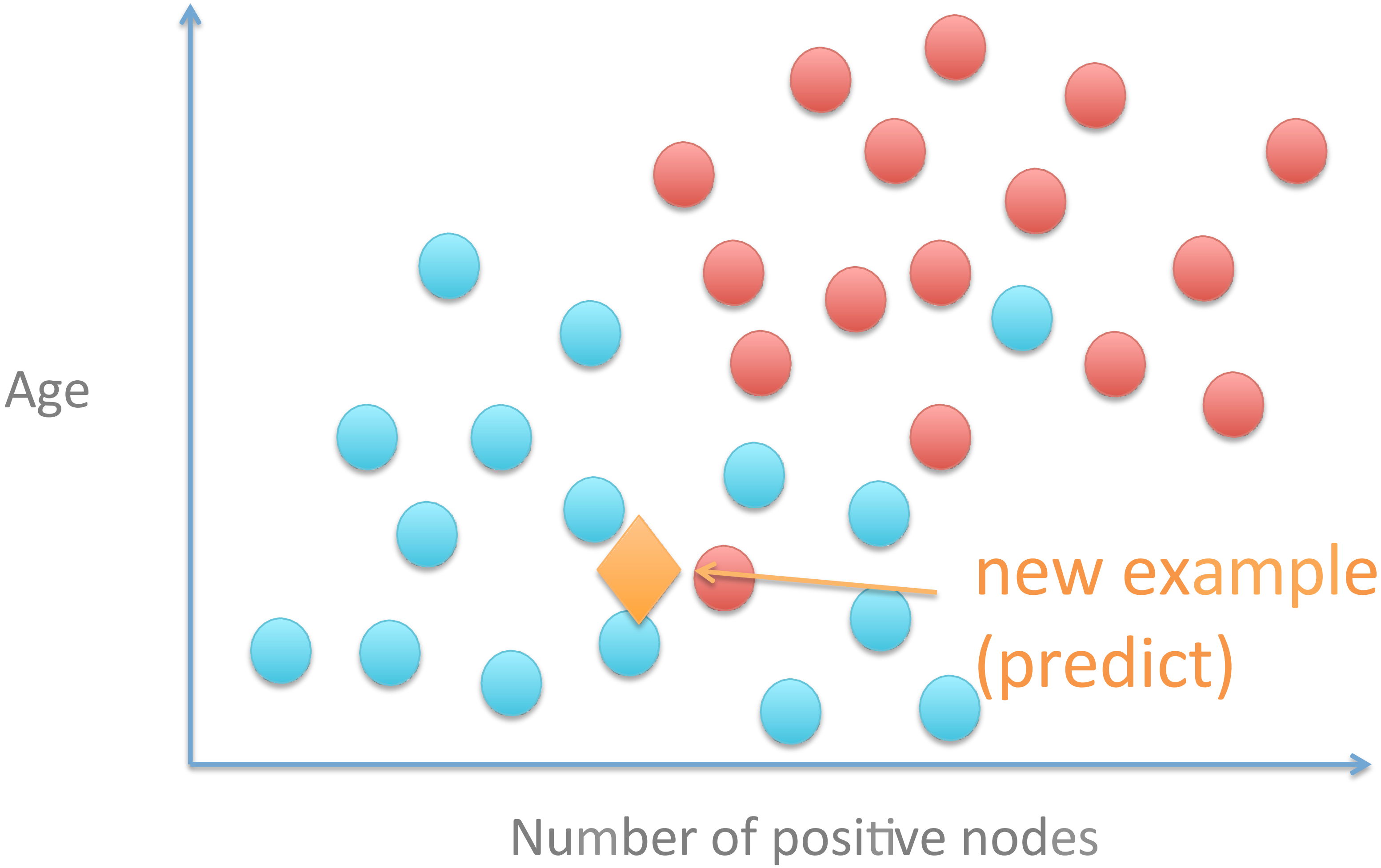
2 Features: Number of + nodes, Age

2 Labels: Survived / Lost

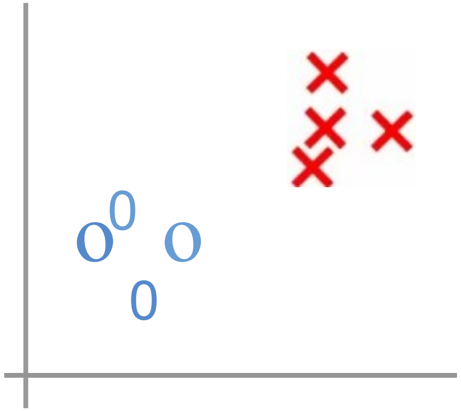


2 Features: Number of + nodes, Age

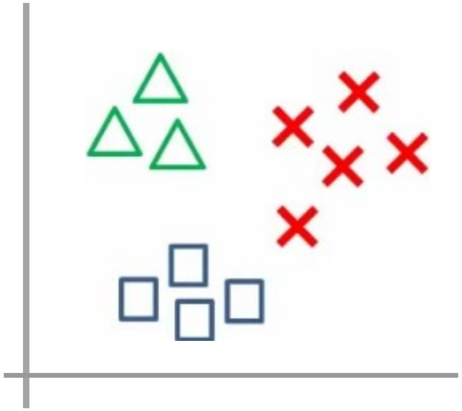
2 Labels: Survived / Lost



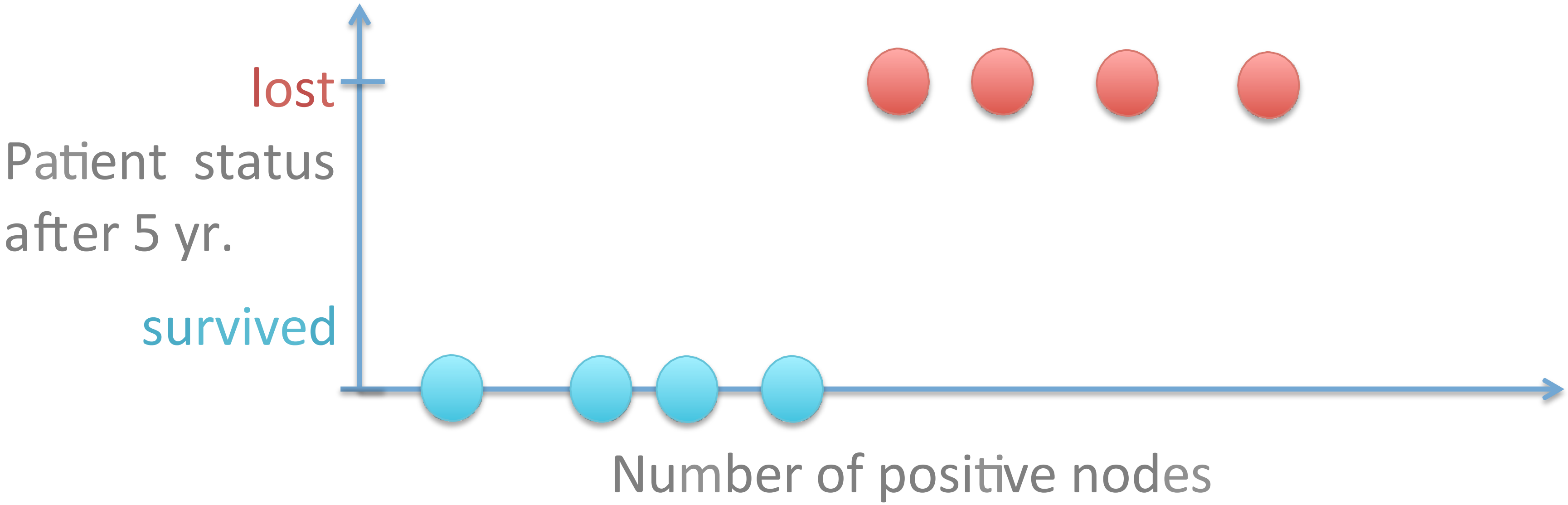
Binary classification:



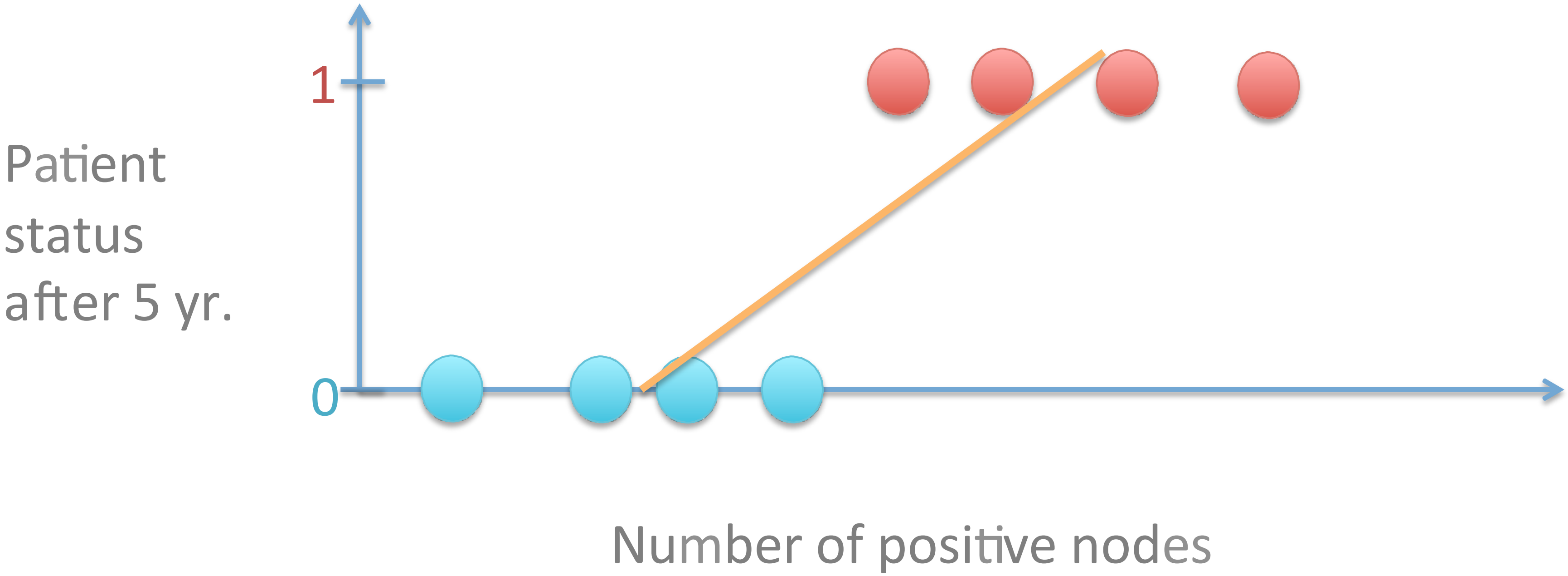
Multi-class classification:



Linear regression for classification?

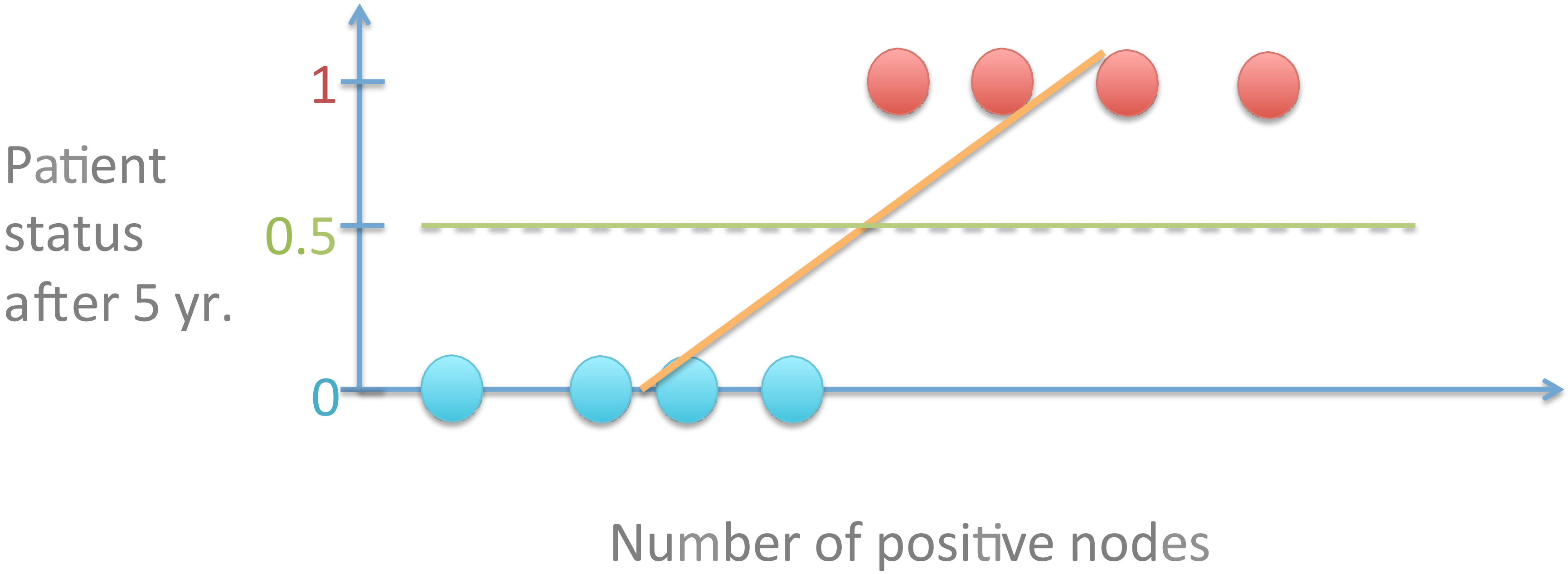


Linear regression for classification?



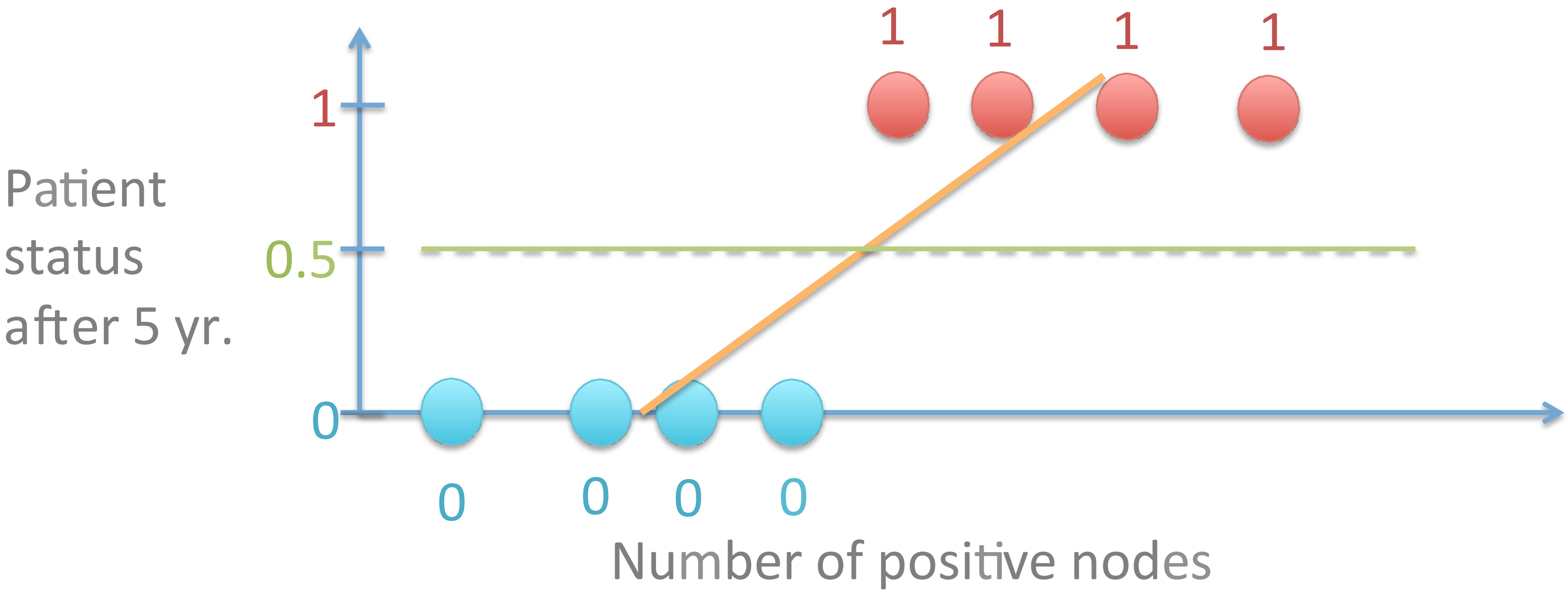
$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$

Linear regression for classification?



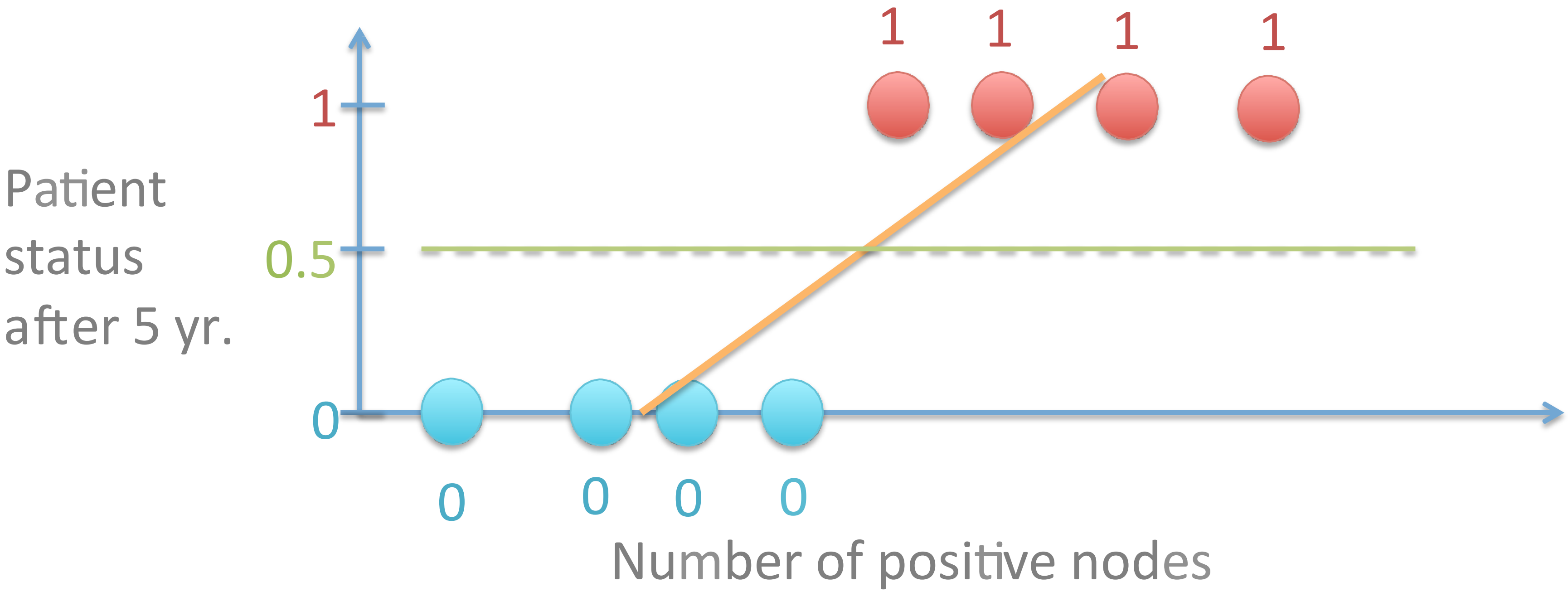
If y_pred > 0.5 predict label 1 (lost)
If y_pred < 0.5 predict label 0 (survived)

Linear regression for classification?



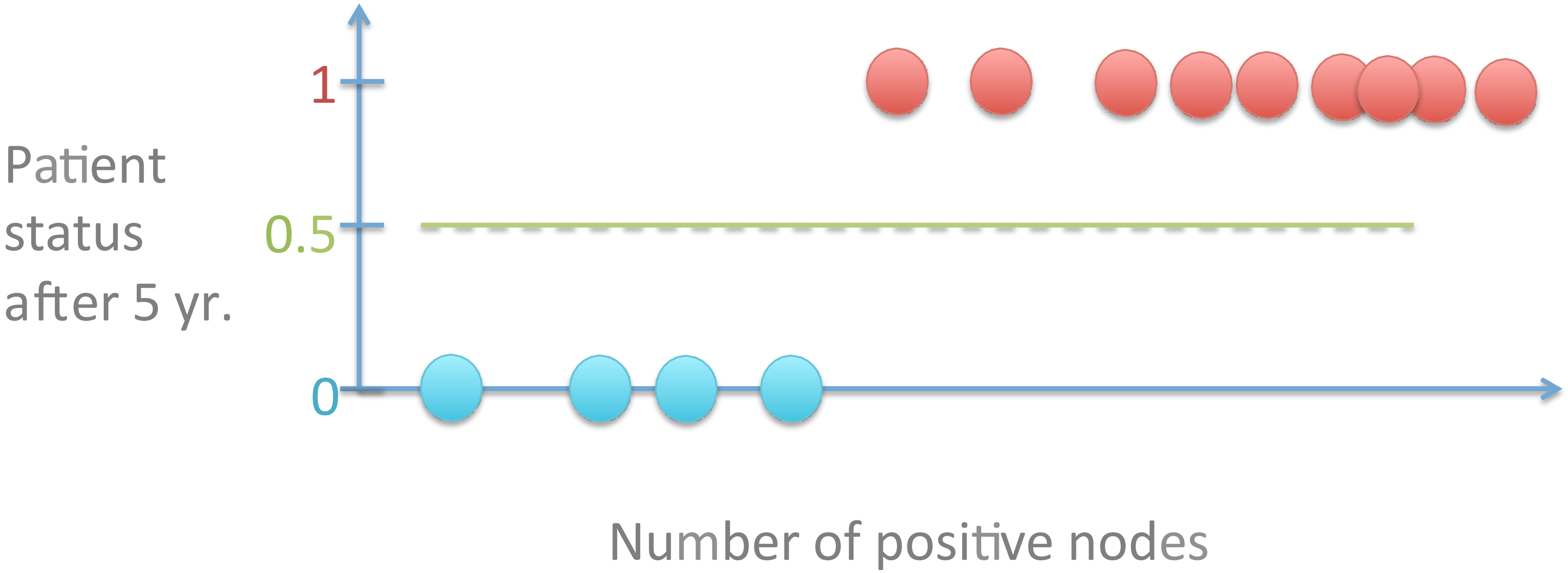
If y_pred > 0.5 predict label 1 (lost)
If y_pred < 0.5 predict label 0 (survived)

Linear regression for classification?



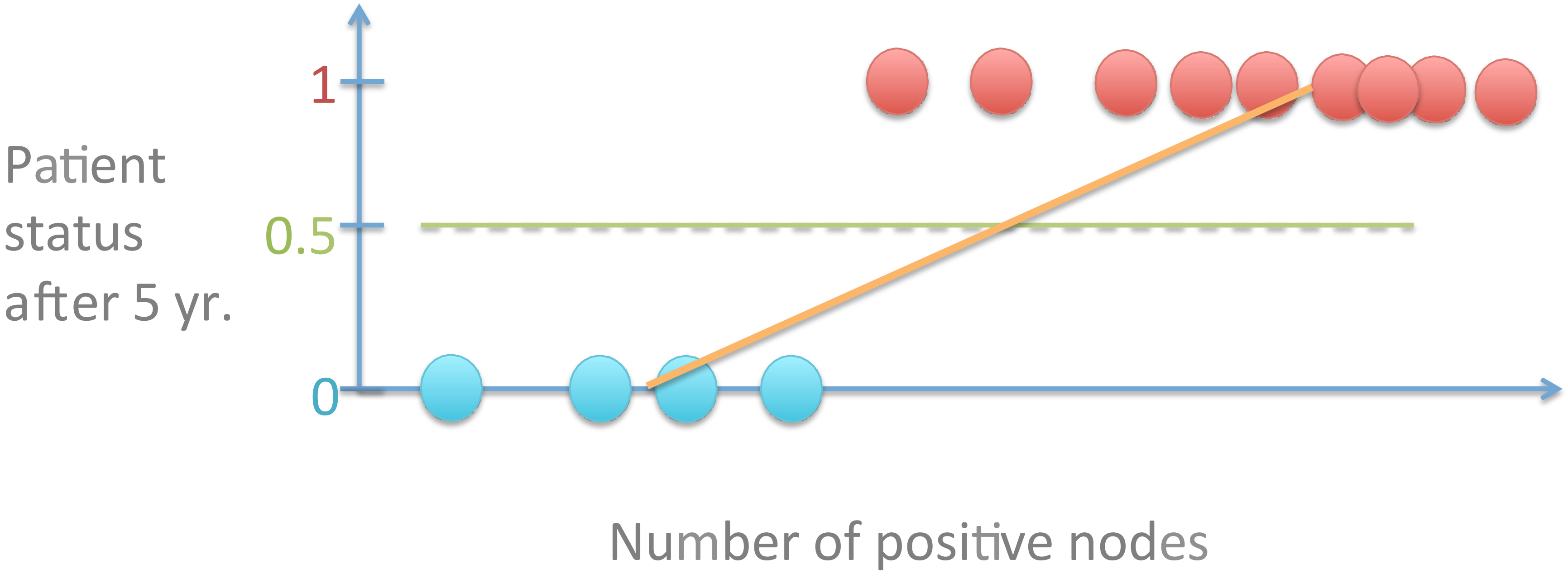
If y_pred > 0.5 predict label 1 (lost)
If y_pred < 0.5 predict label 0 (survived)

Linear regression for classification?



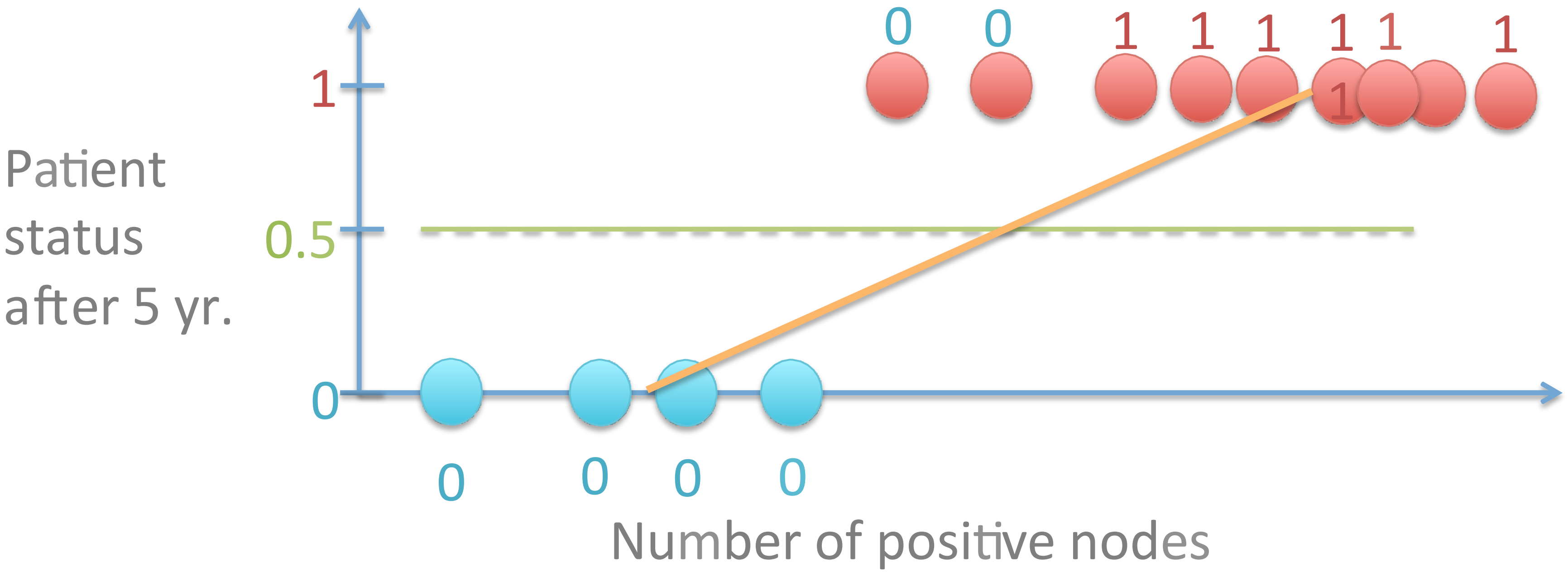
If y_pred > 0.5 predict label 1 (lost)
If y_pred < 0.5 predict label 0 (survived)

Linear regression for classification?



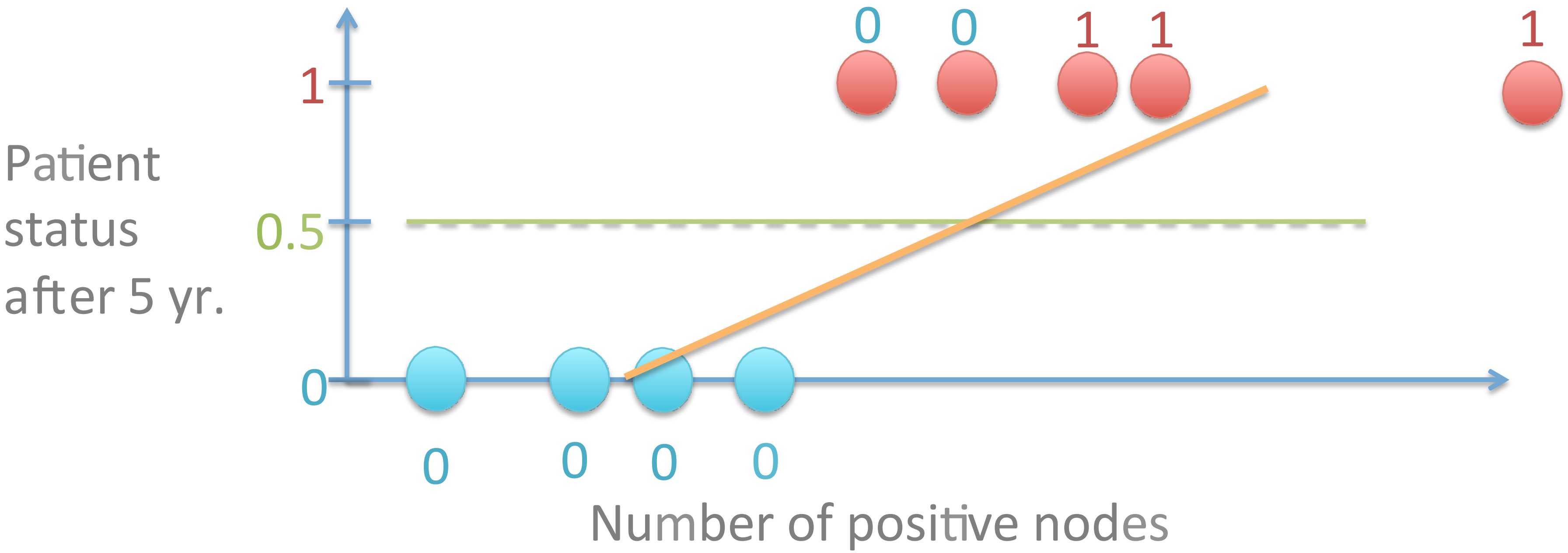
If y_pred > 0.5 predict label 1 (lost)
If y_pred < 0.5 predict label 0 (survived)

Linear regression for classification?



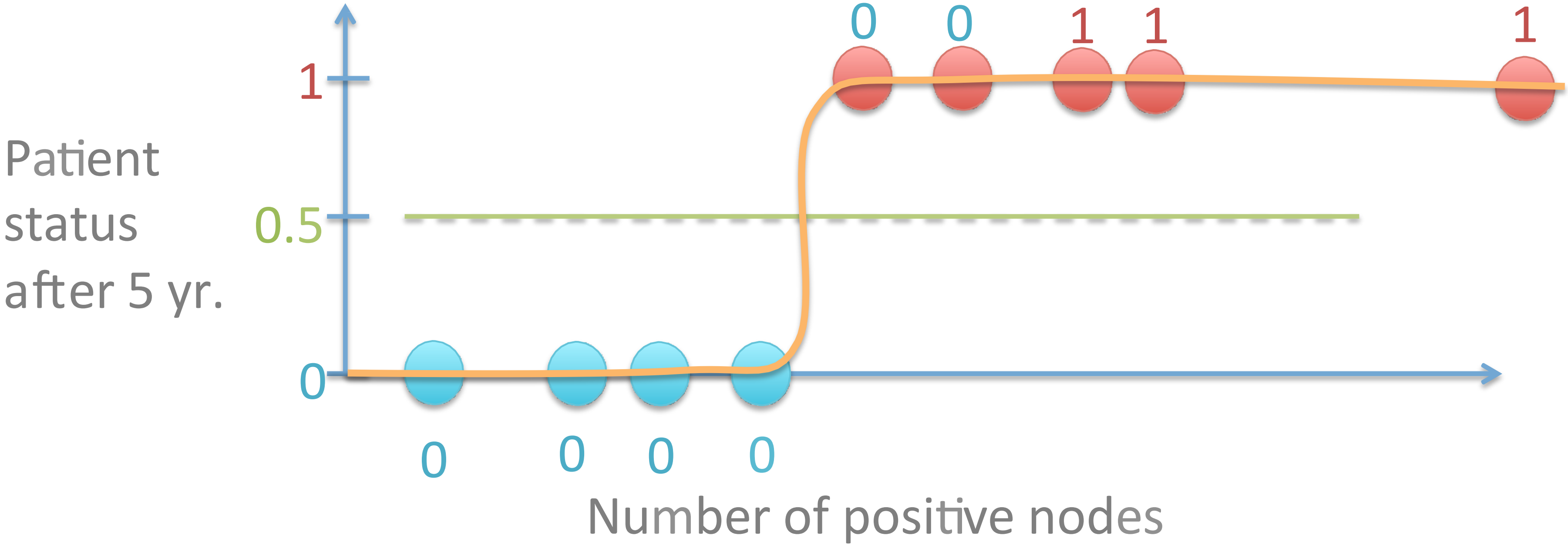
If y_pred > 0.5 predict label 1 (lost)
If y_pred < 0.5 predict label 0 (survived)

Linear regression for classification?



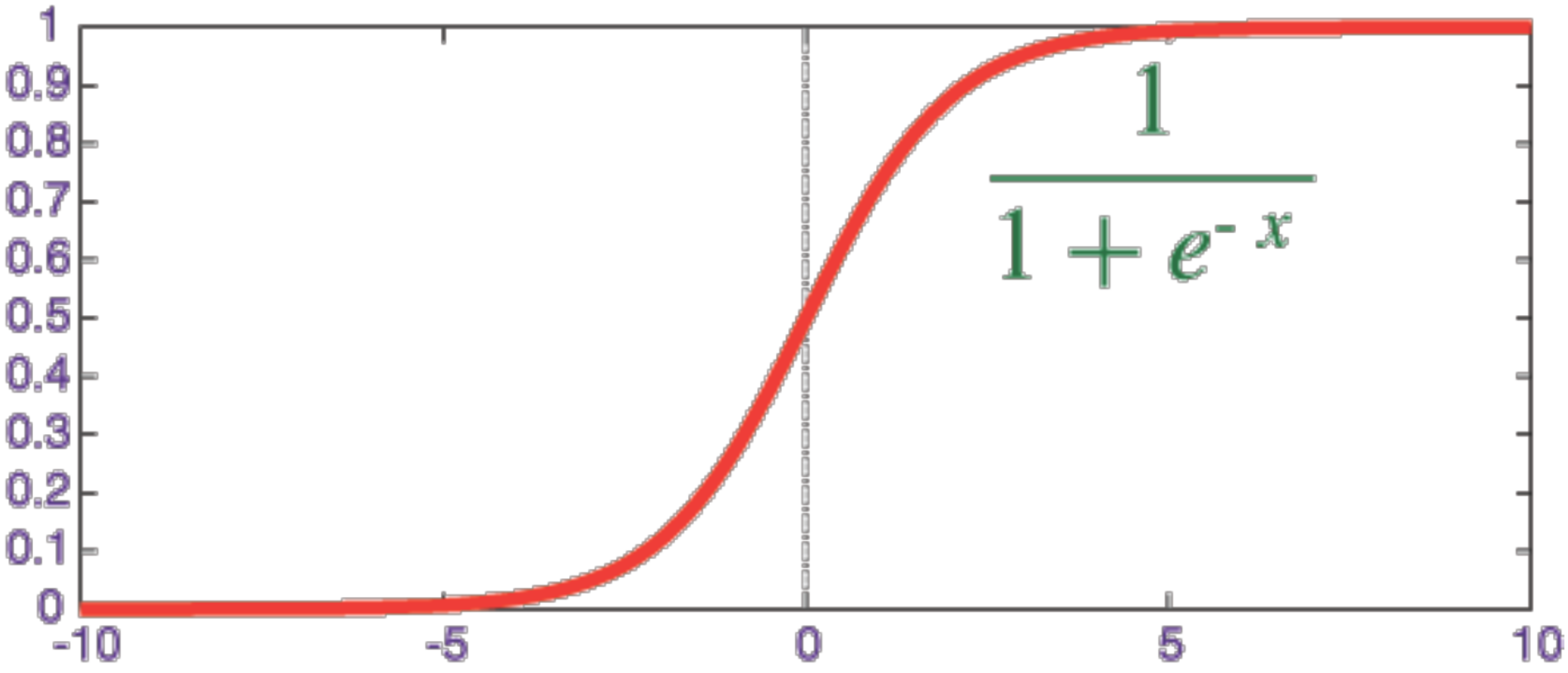
If y_pred > 0.5 predict label 1 (lost)
If y_pred < 0.5 predict label 0 (survived)

Logistic regression to the rescue

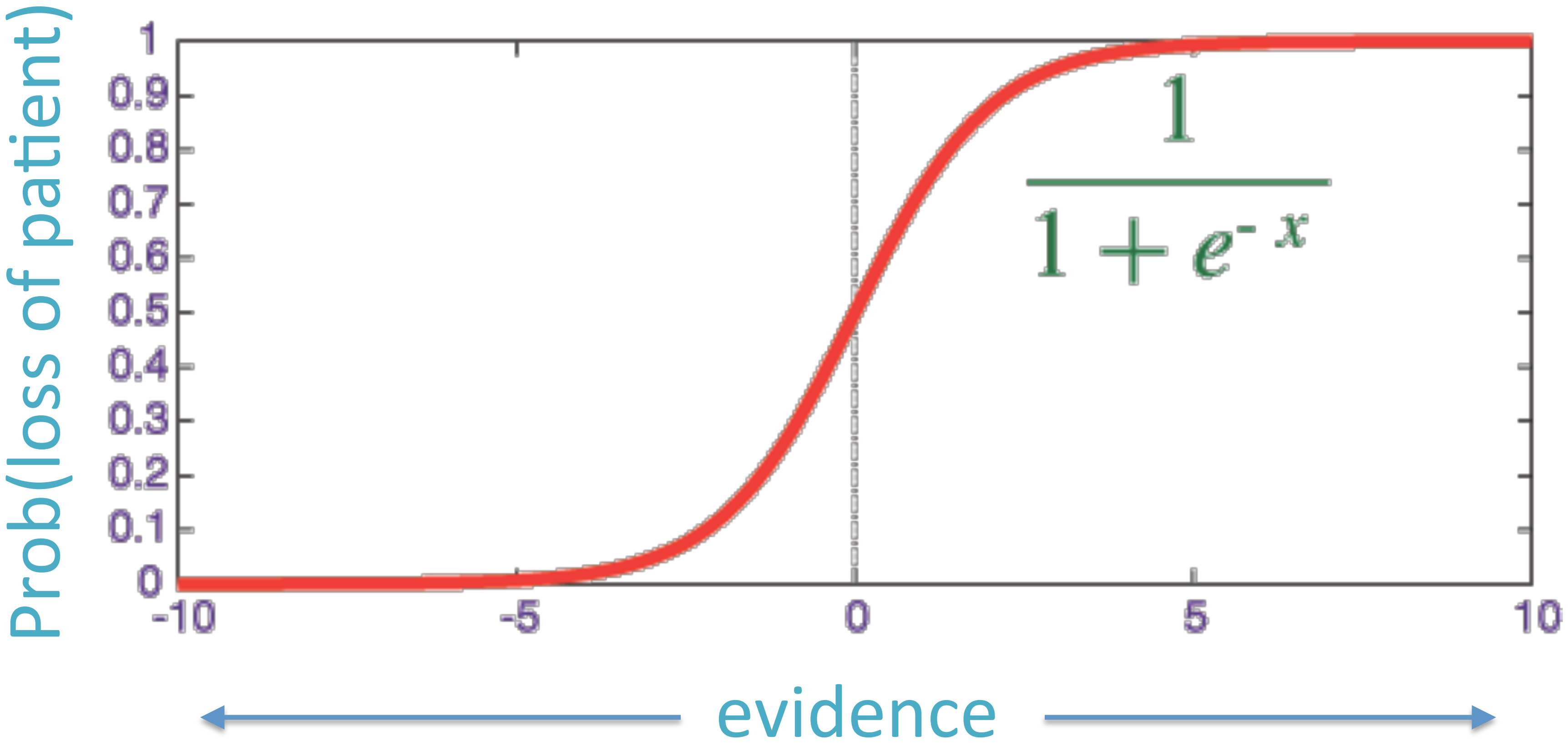


$$y_{\beta}(x) = \frac{1}{1+e^{-(\beta_0+\beta_1 x+\varepsilon)}}$$

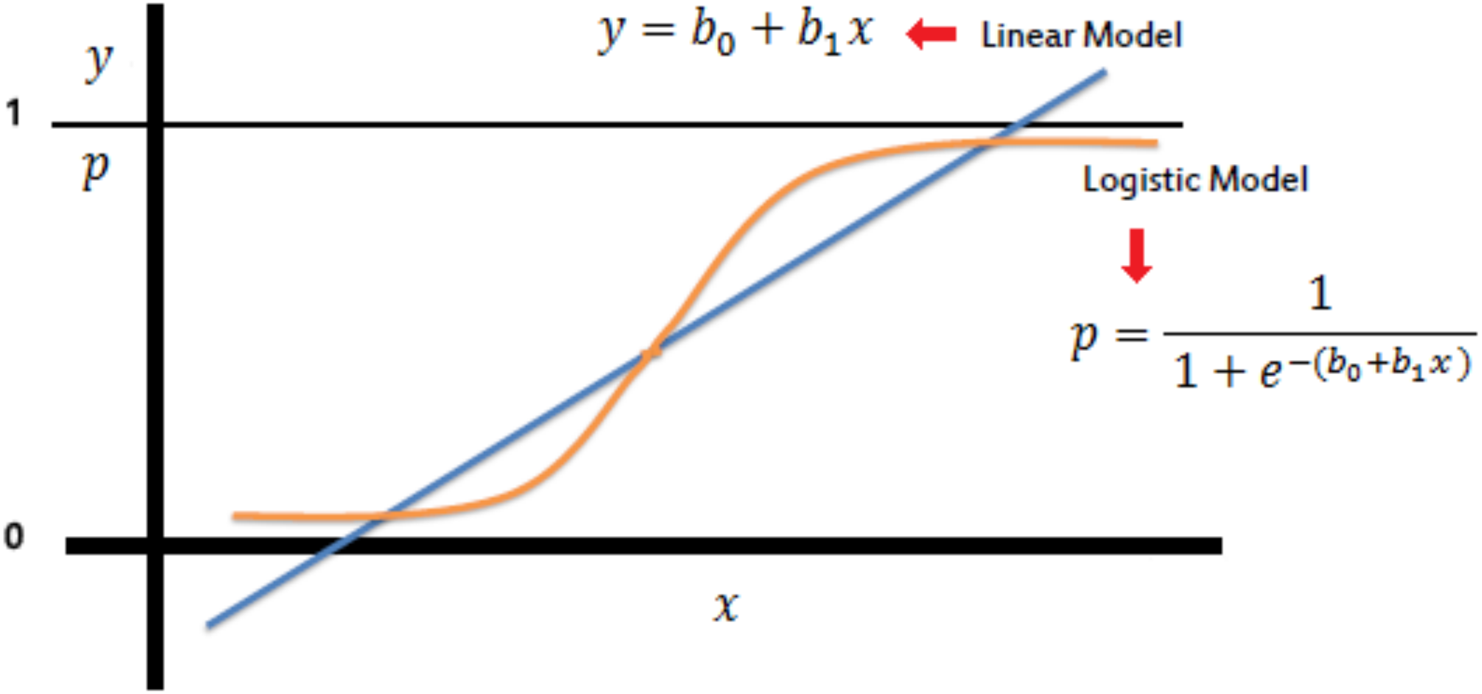
What is this function?



What is this function?

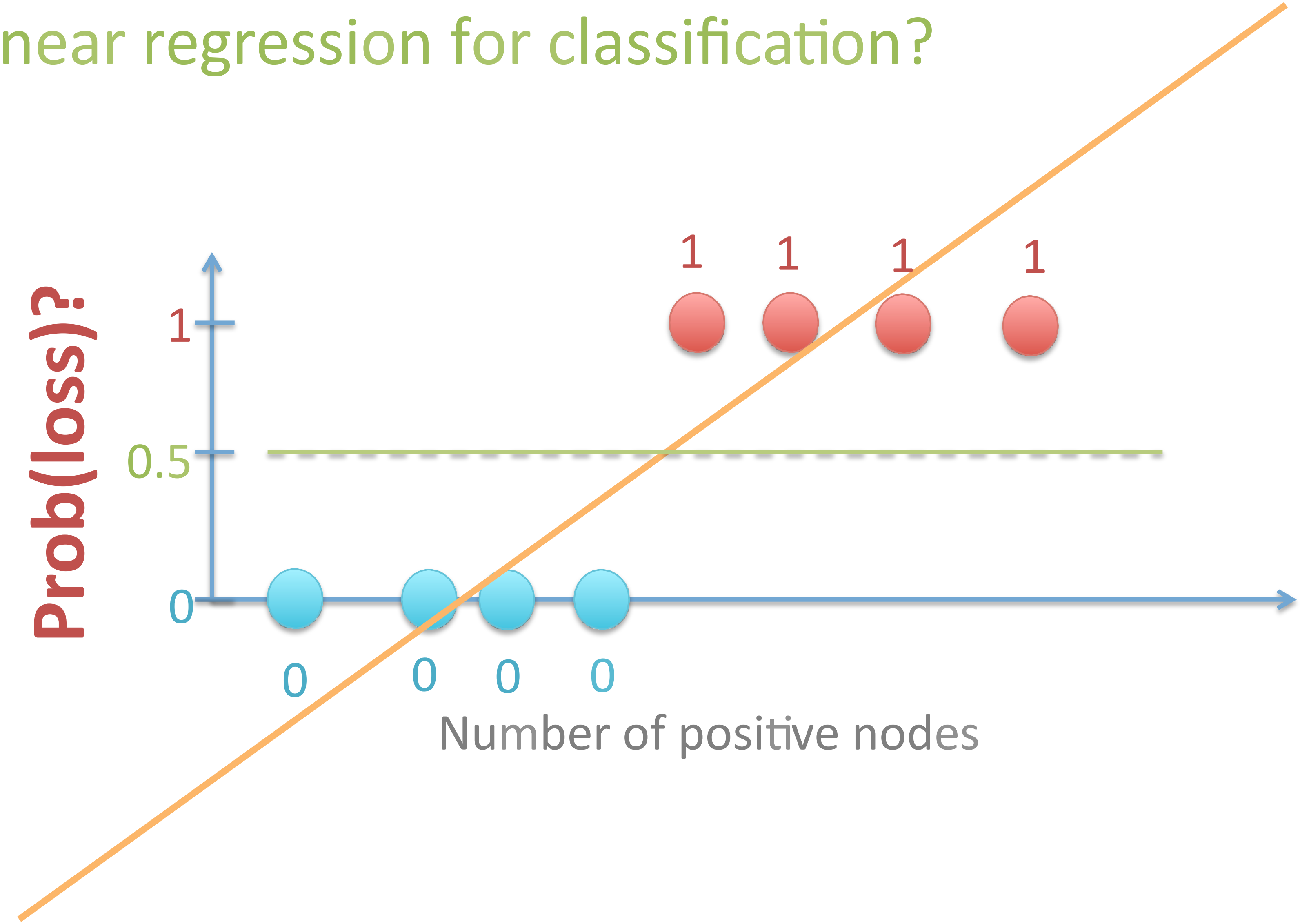


What is this function?

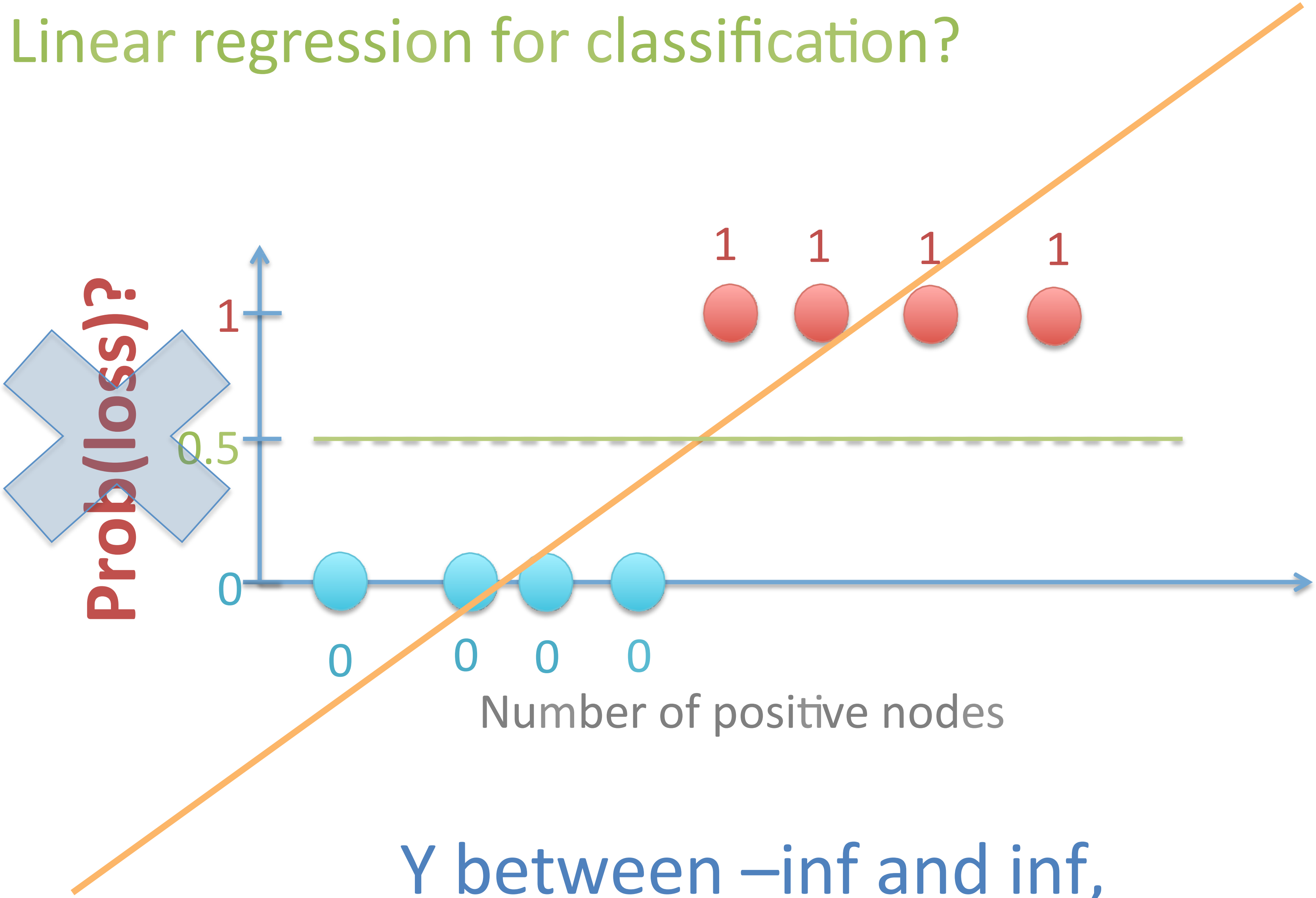


http://www.saedsayad.com/logistic_regression.htm

Linear regression for classification?



Linear regression for classification?



Y between –inf and inf,
Not 0 and 1

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.8$$

$$P(survival) = 0.2$$

Probability

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.8$$

$$P(survival) = 0.2$$

$$\frac{P(loss)}{P(survival)} = 4$$

Probability

Odds

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.05$$

$$P(survival) = 0.95$$

$$\frac{P(loss)}{P(survival)} = 0.053$$

Probability

Odds

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.5$$

$$P(survival) = 0.5$$

$$\frac{P(loss)}{P(survival)} = 1$$

Probability

Odds

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.5$$

$$P(survival) = 0.5$$

Probability

$$\frac{P(loss)}{P(survival)} = 1$$

Odds

between 0 and inf

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.5$$

$$P(survival) = 0.5$$

Probability

$$\log\left(\frac{P(loss)}{P(survival)}\right) = 0$$

Log Odds

between -inf and inf

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.05$$

$$P(survival) = 0.95$$

$$\log\left(\frac{P(loss)}{P(survival)}\right) = -2.94$$

Probability

Log Odds

between -inf and inf

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.8$$

$$P(survival) = 0.2$$

Probability

$$\log\left(\frac{P(loss)}{P(survival)}\right) = 1.39$$

Log Odds

between -inf and inf

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.999$$

$$P(survival) = 0.001$$

$$\log\left(\frac{P(loss)}{P(survival)}\right) = 6.9$$

Probability

Log Odds

between -inf and inf

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.999$$

$$1 - P(loss) = 0.001$$

$$\log\left(\frac{P(loss)}{1 - P(loss)}\right) = 6.9$$

Probability

Log Odds

logit function

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.999$$

$$1 - P(loss) = 0.001$$

Probability

$$\log\left(\frac{P(loss)}{1 - P(loss)}\right) = 6.9$$

Log Odds

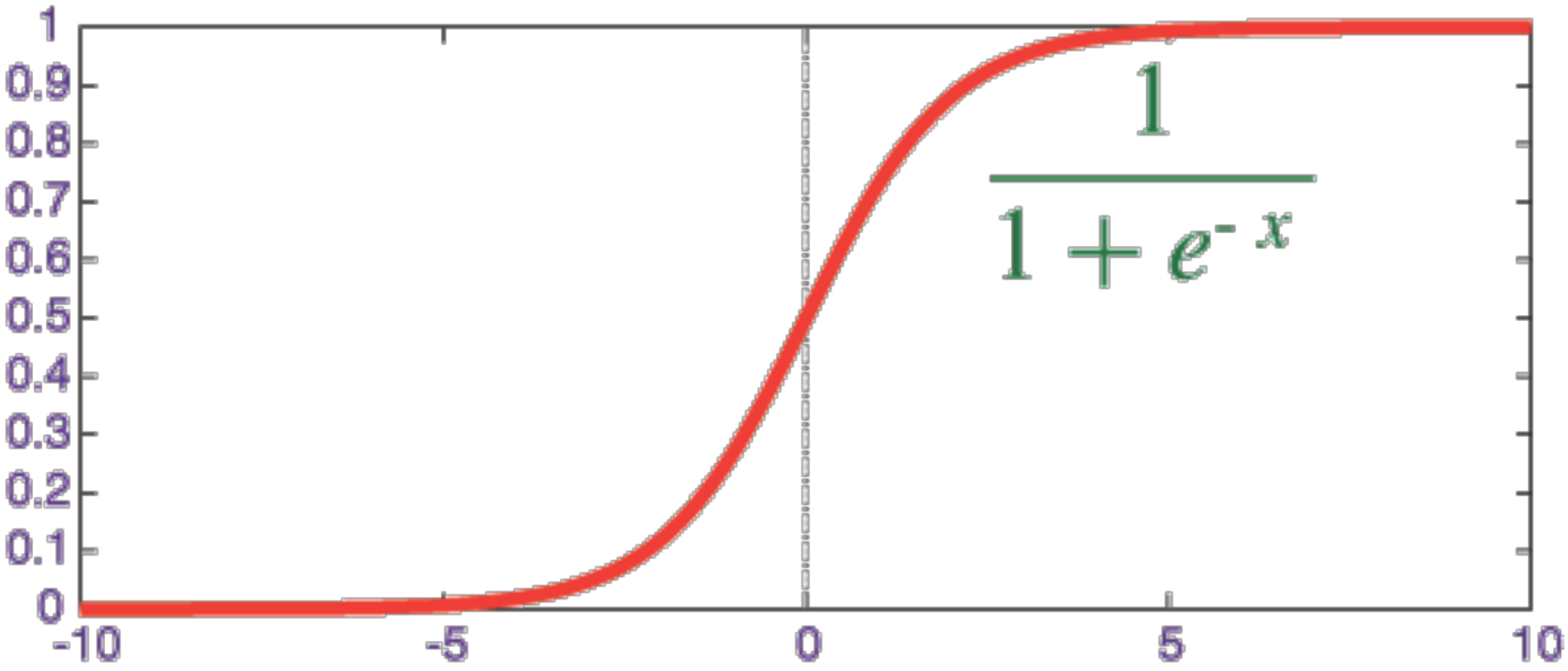
logit function

$$\frac{1}{1 - e^{\log\left(\frac{P(loss)}{1-P(loss)}\right)}} = P(loss)$$

Logistic Function

Log Odds → Prob

What is this function?



Prob(loss of patient)

evidence

“Logistic regression” is a classification algorithm

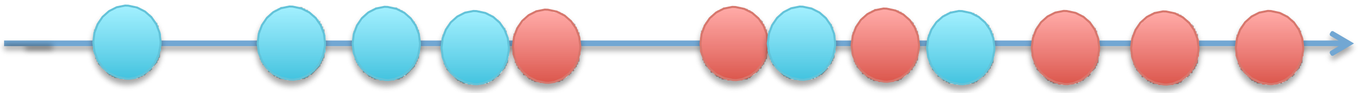
$$y_{\beta}(x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x + \varepsilon)}}$$

```
from sklearn.linear_model import LogisticRegression
#(just like LinearRegression)
```

```
from statsmodels.formula.api import Logit
#(just like OLS)
```

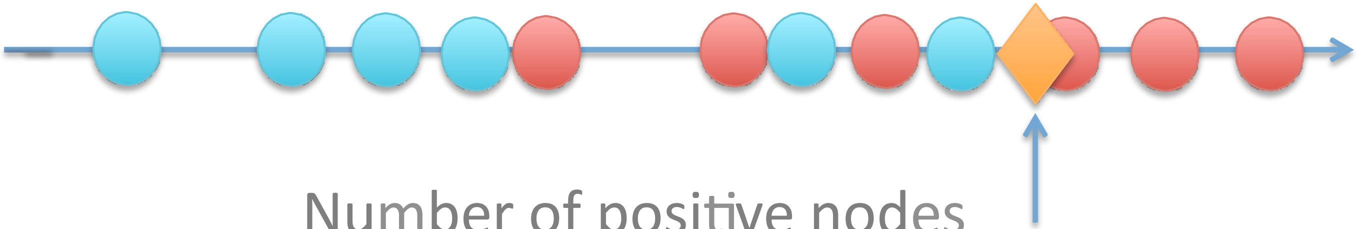
K Nearest Neighbors

K Nearest Neighbors



Number of positive nodes

K Nearest Neighbors

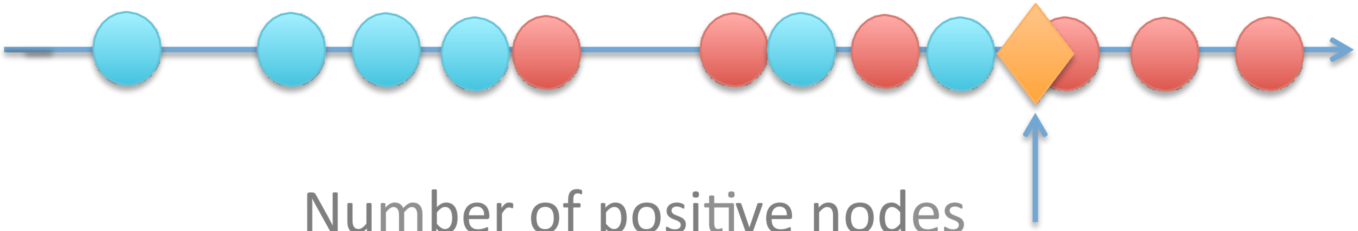


Number of positive nodes

new example (predict)

K Nearest Neighbors

K=1:
Look at the nearest neighbor, predict their label

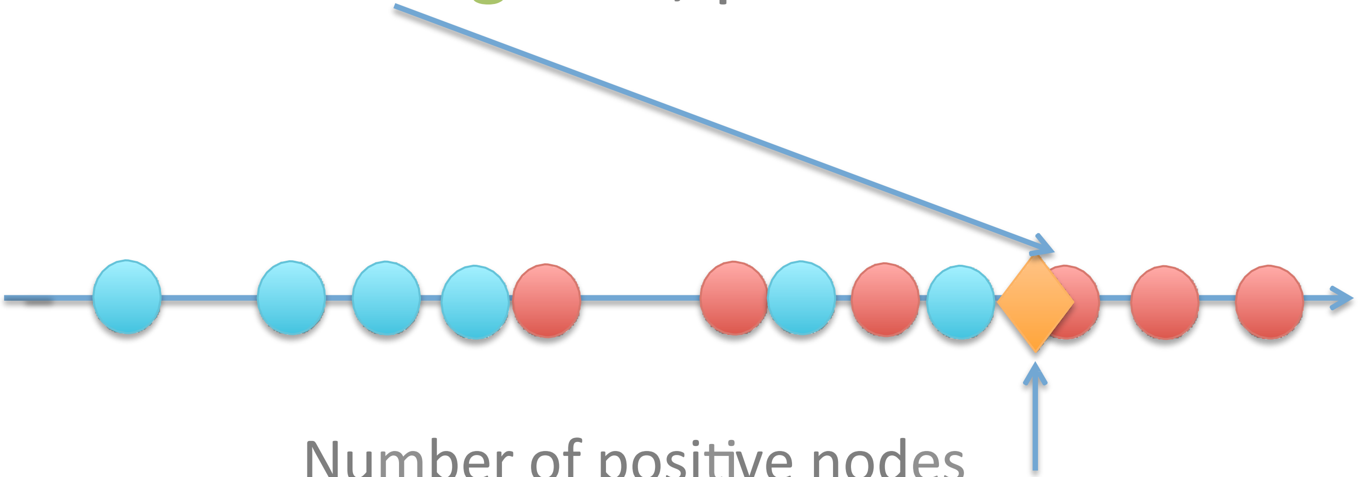

Number of positive nodes

new example (predict)

K Nearest Neighbors

K=1:

Look at the nearest neighbor, predict their label



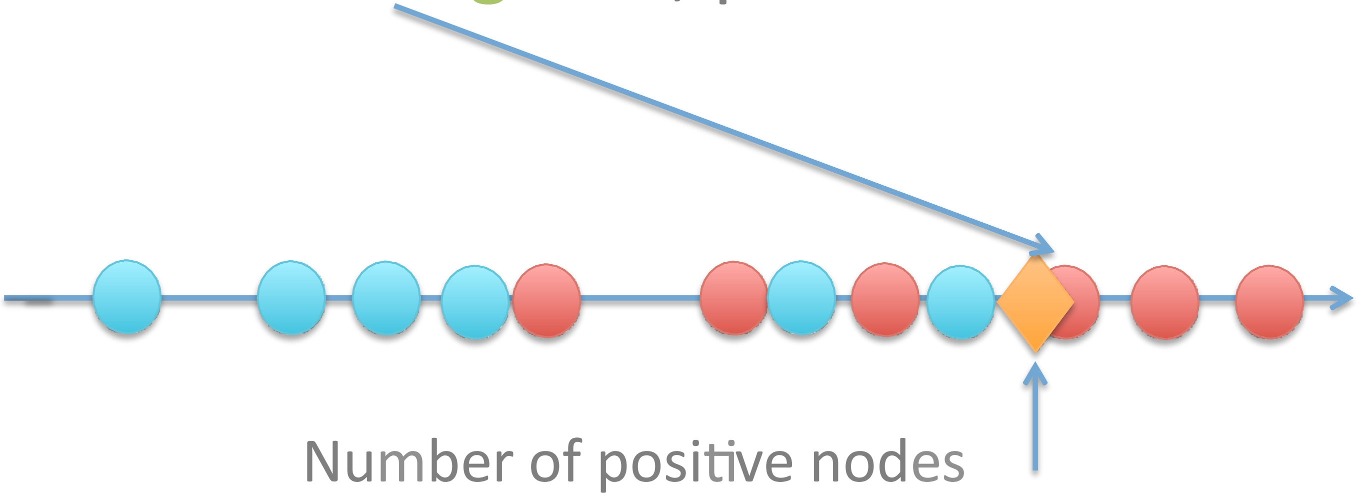
Number of positive nodes

new example (predict)

K Nearest Neighbors

K=1:

Look at the nearest neighbor, predict their label



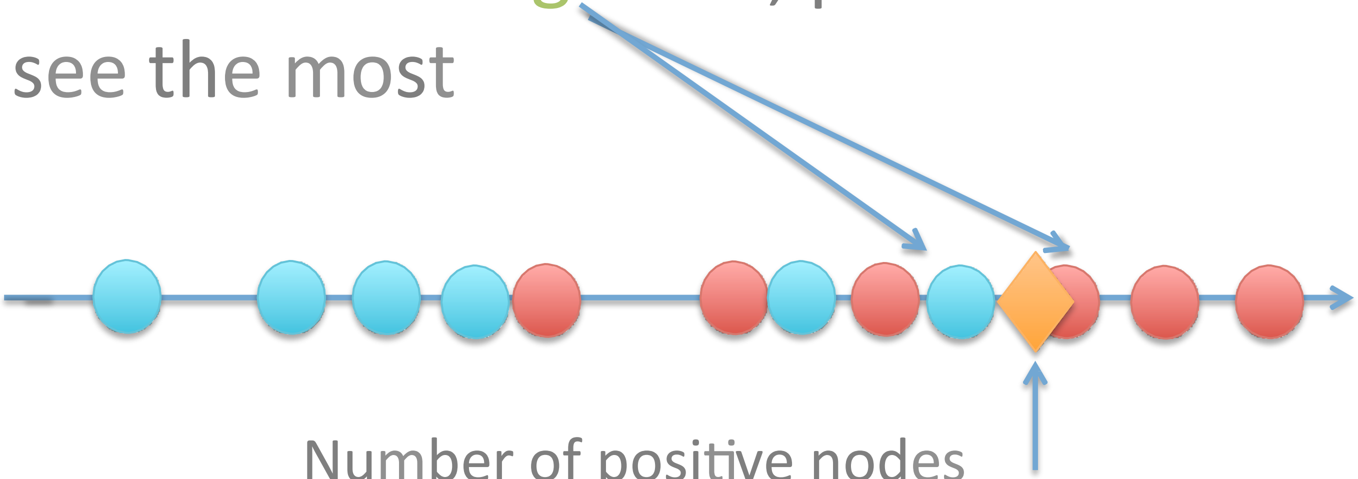
new example

predict: 🔴

K Nearest Neighbors

K=2:

Look at the 2 nearest neighbors, predict the label you see the most



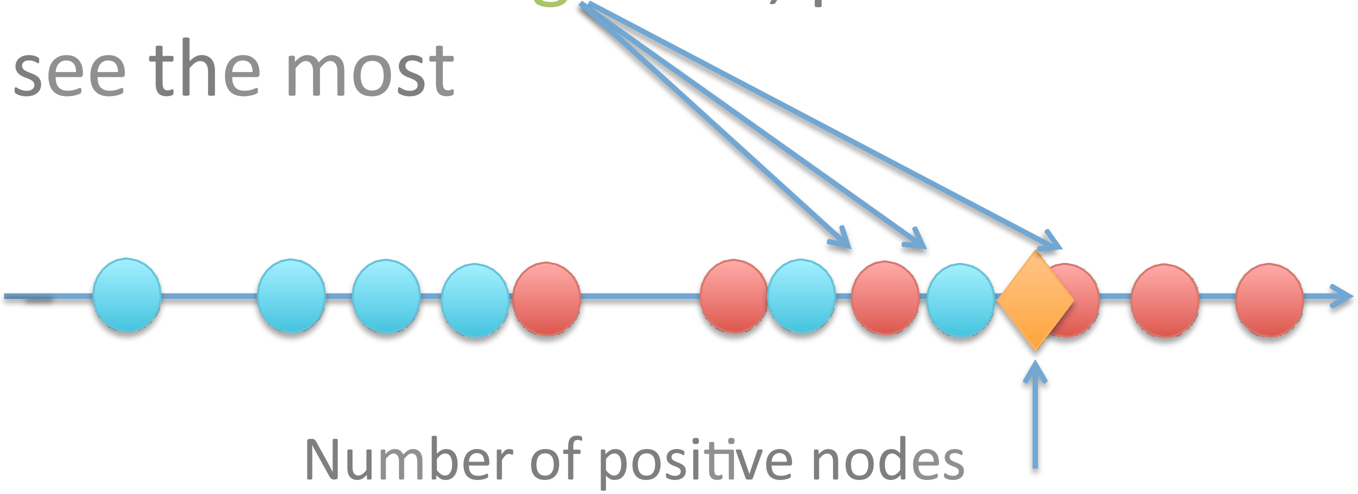
new example

predict: ●

K Nearest Neighbors

K=3:

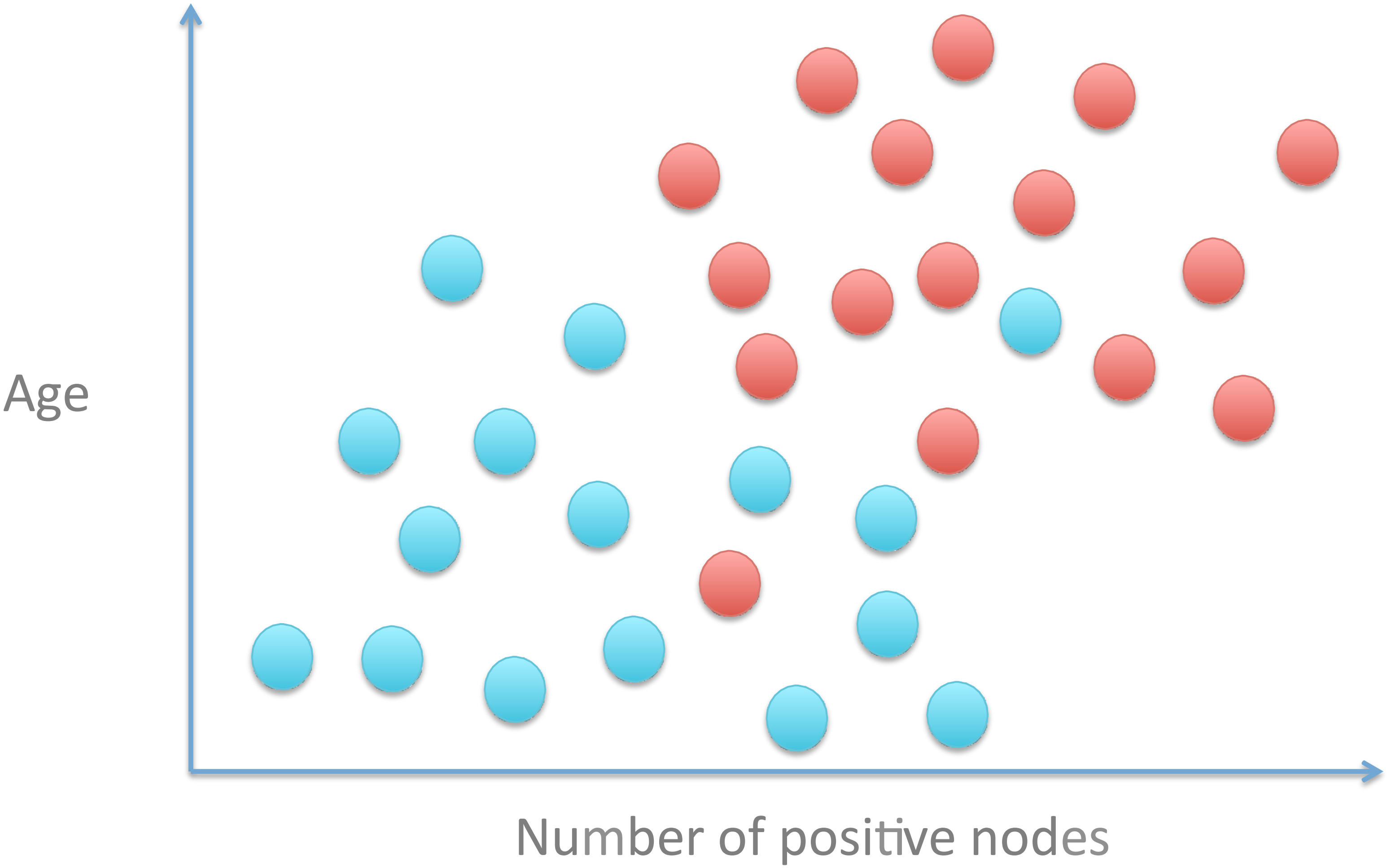
Look at the 3 nearest neighbors, predict the label you see the most



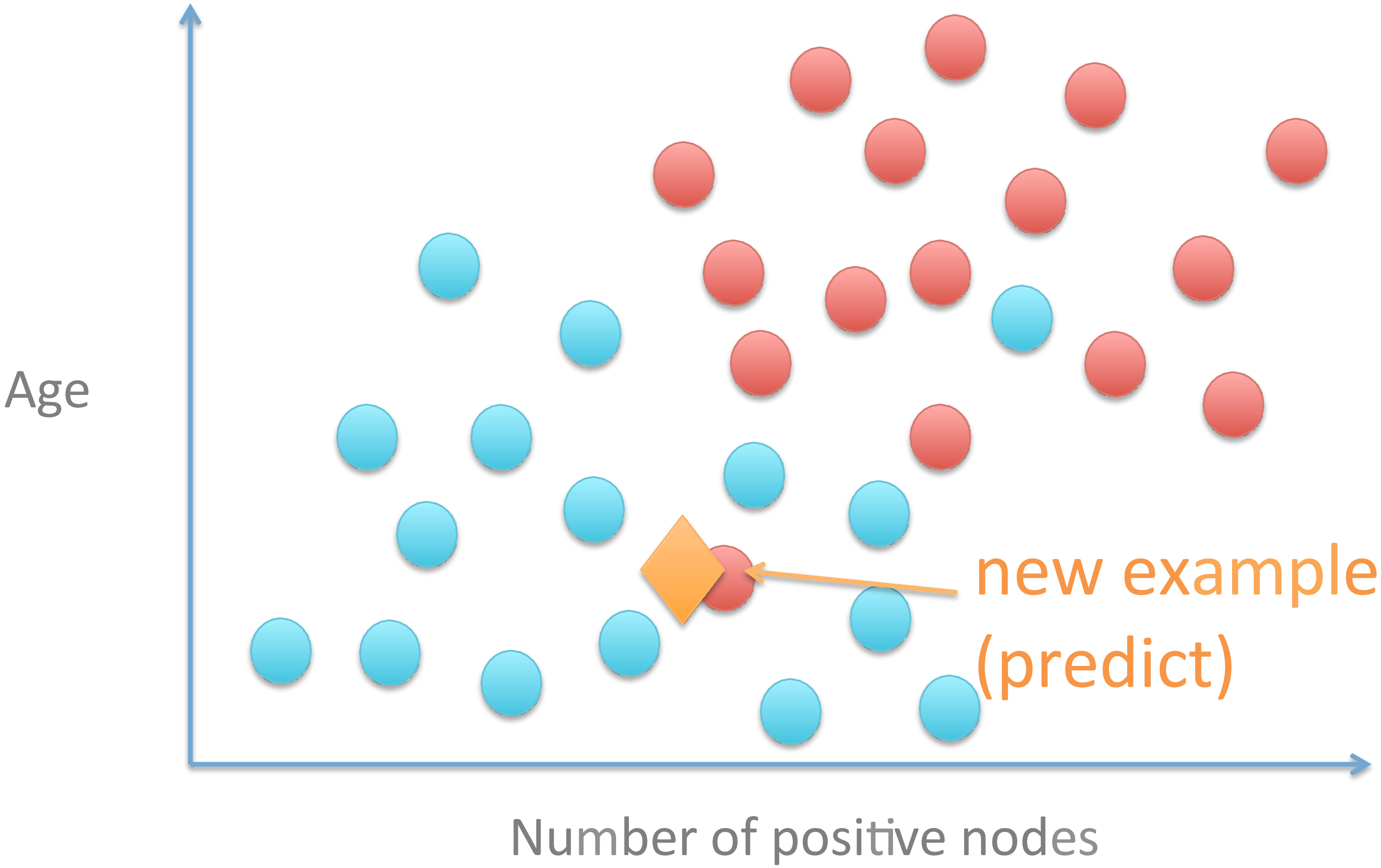
new example

predict: ●

K Nearest Neighbors



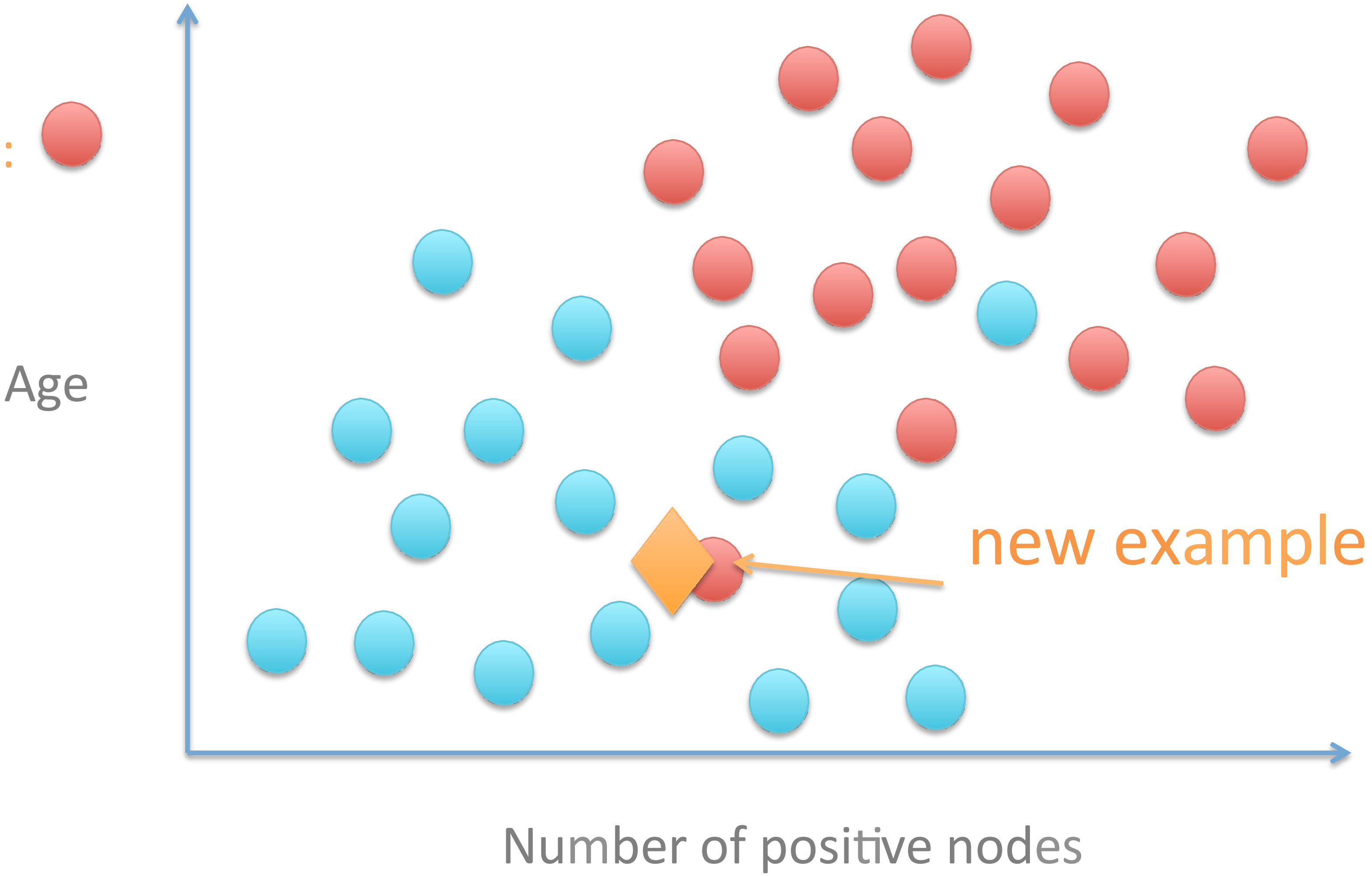
K Nearest Neighbors



K Nearest Neighbors

K=1

Predict: 🔴

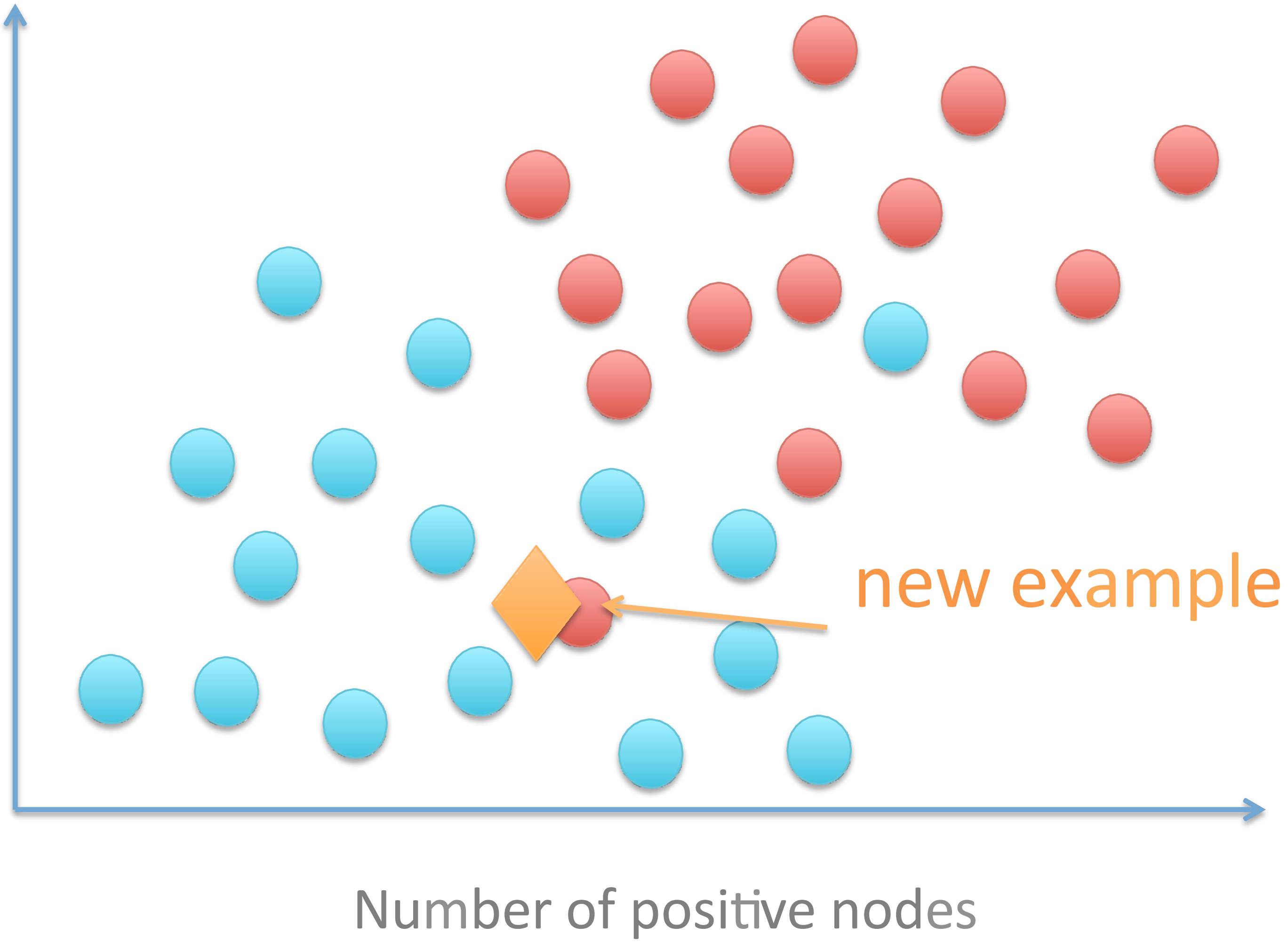


new example

K Nearest Neighbors

K=2

Predict: 🔴

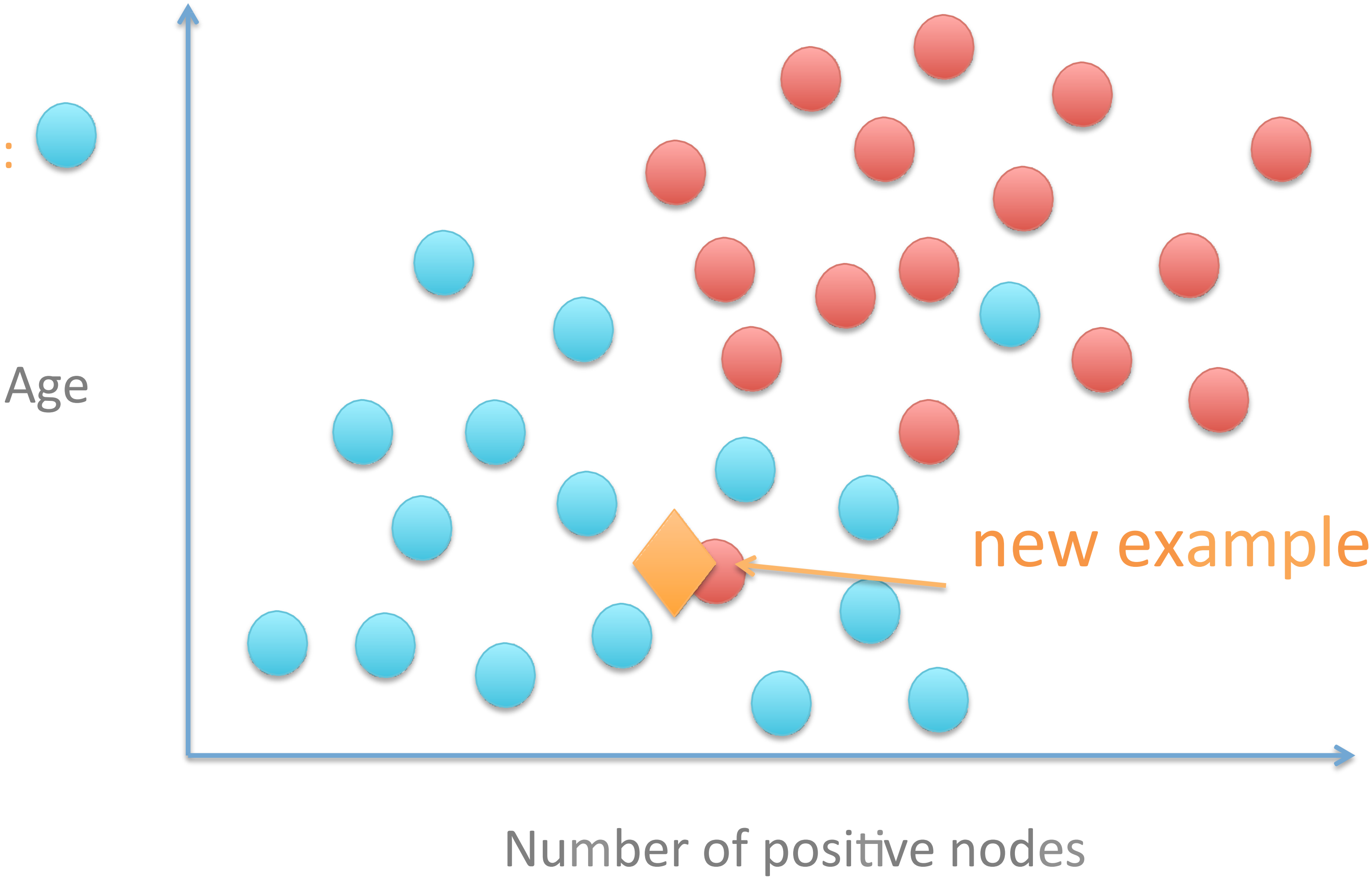


new example

K Nearest Neighbors

K=3

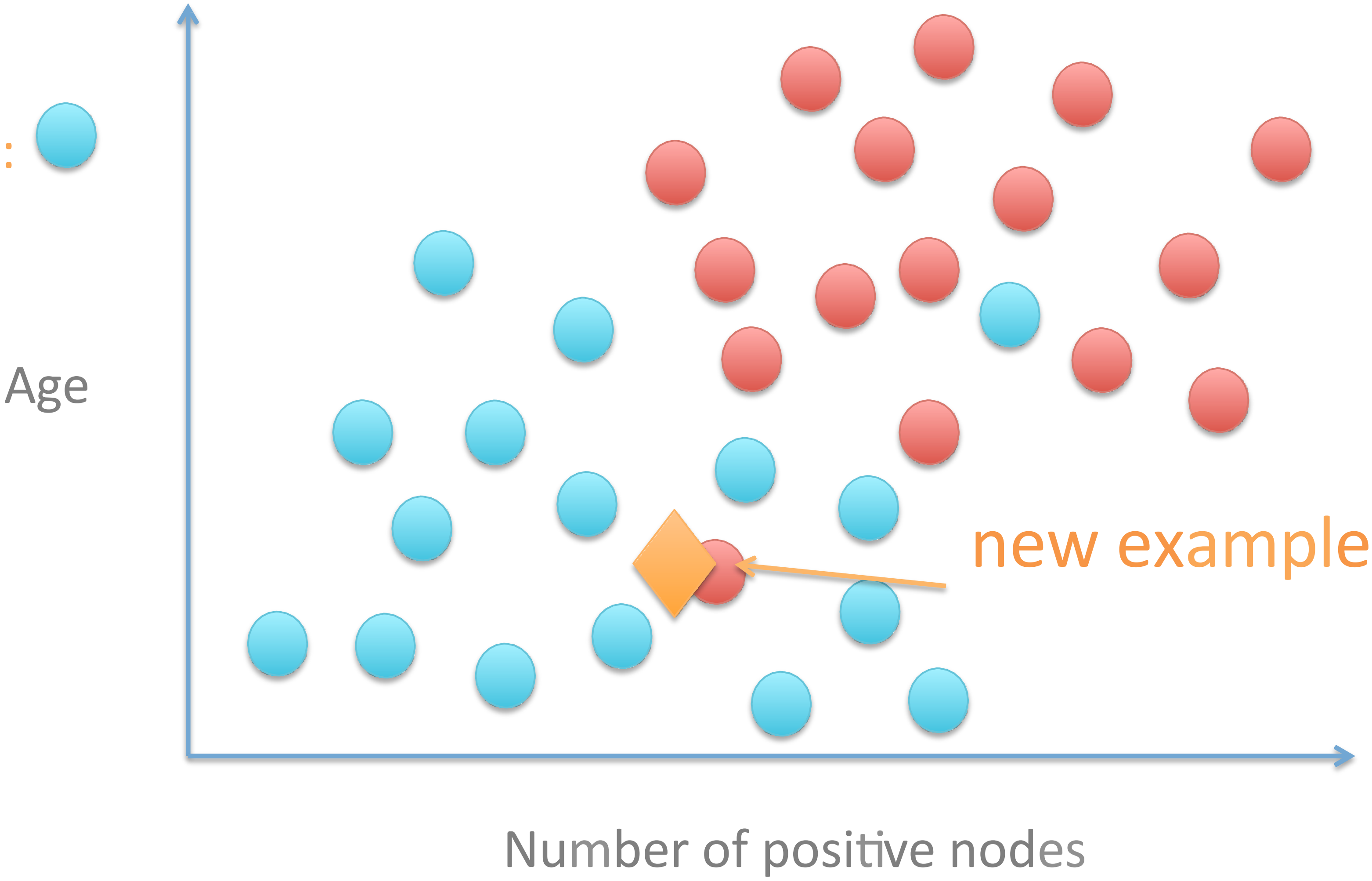
Predict: ●



K Nearest Neighbors

K=4

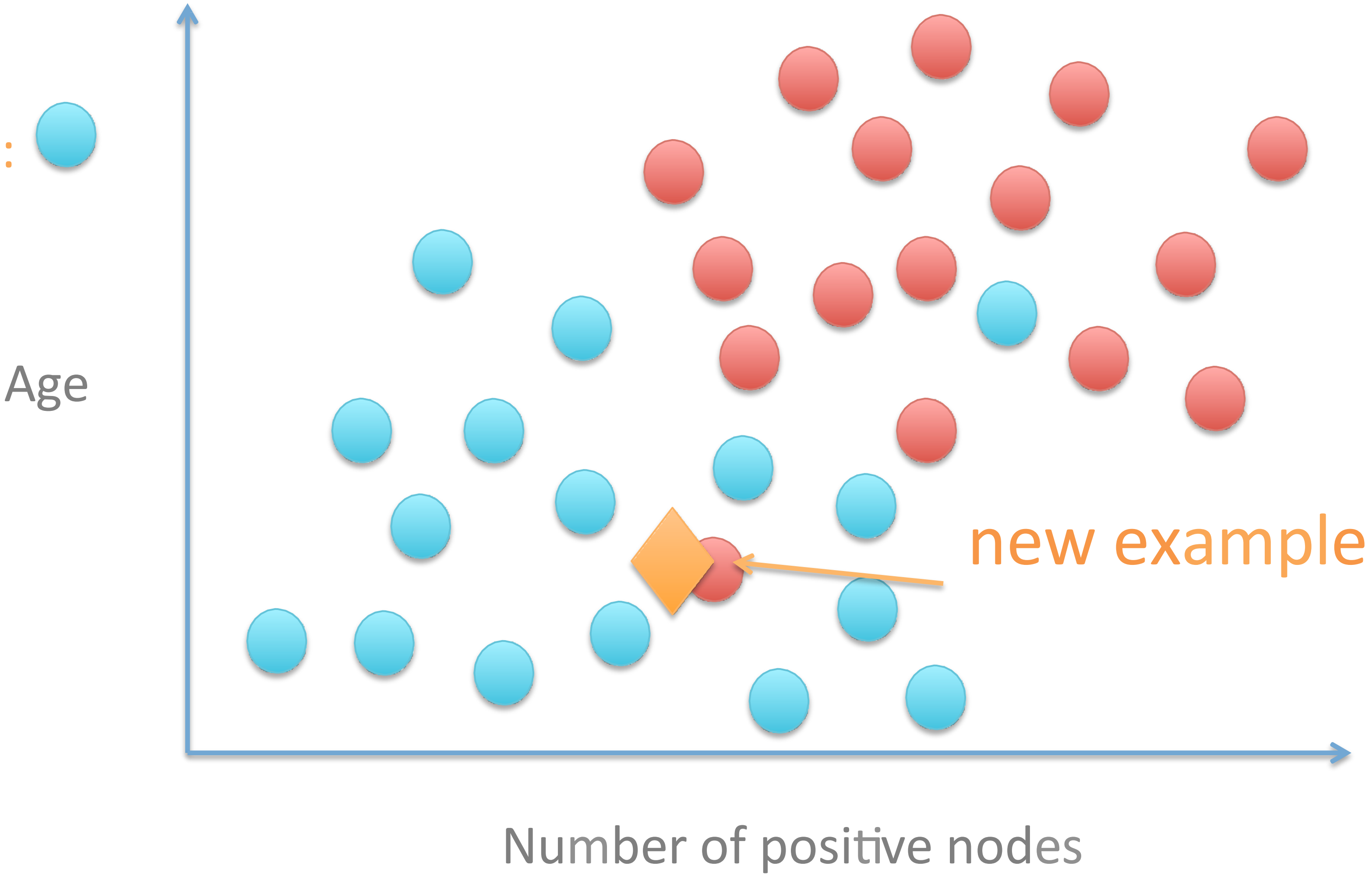
Predict: ●

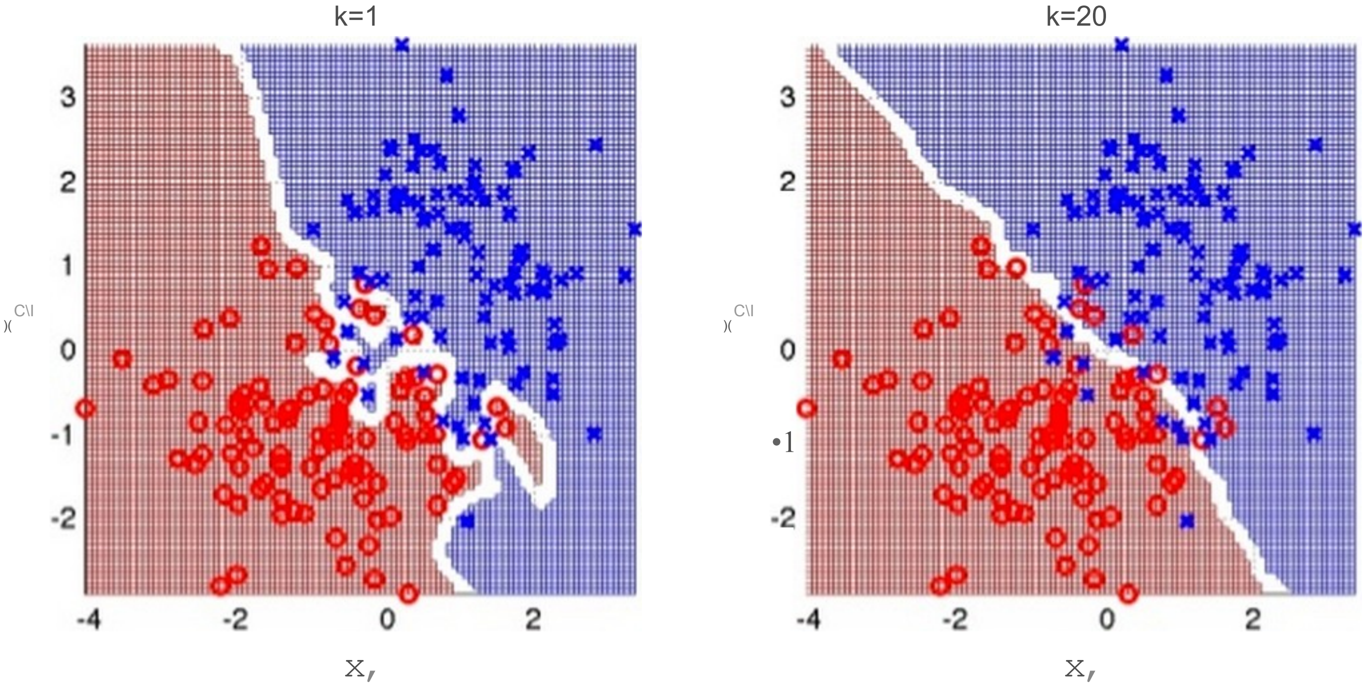


K Nearest Neighbors

K=5

Predict: ●




k=1
3
2
1
)(C\l
0
-1
-2
-4
-2
0
2
x,
k=20
3
2
1
)(C\l
0
•1
-2
-4
-2
0
2
x,

```
from sklearn.neighbors import KNeighborsClassifier
```