Biostat 202: Opportunities and Challenges of "Big Data".

Machine Learning 2: Supervised learning [classification]

Aaron Wolfe Scheffler

Division of Biostatistics

Department of Epidemiology and Biostatistics

Review of last lecture

- **Goal**: We are not trying to create AI, but rather use some machine learning algorithms for the **purpose of prediction or identifying patterns in data.**

1. Data

2. Train/fit model

Algorithm(s)

3. Output

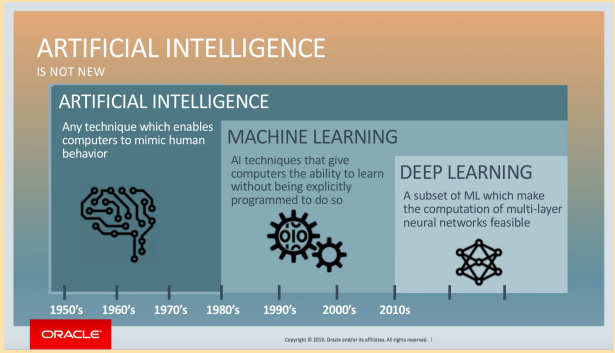
- Prediction
- Dimension Reduction
- Clustering

4. Interpret

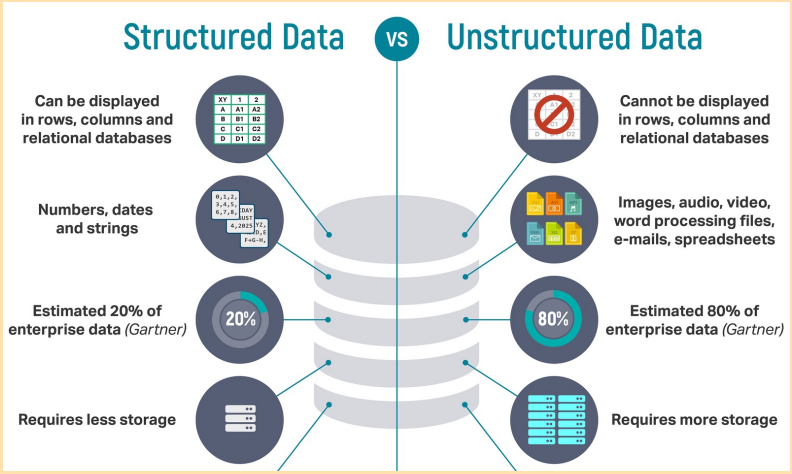
= INFORMATION?

Review of last lecture

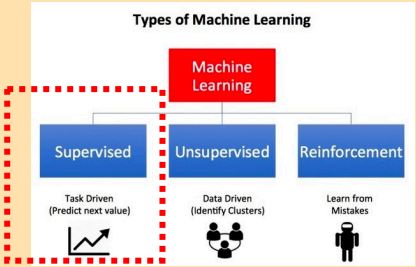
1. Machine learning definition



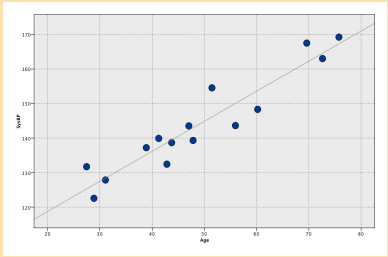
2. Types of data



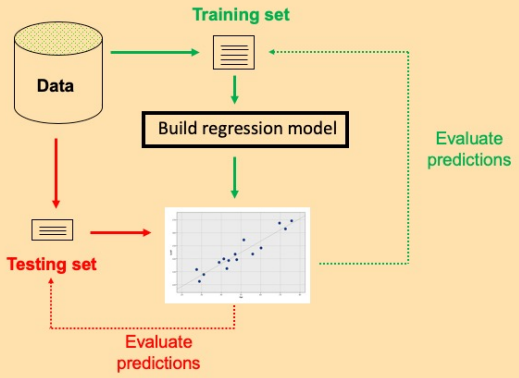
3. Types of machine learning



4. Regression [supervised]



5. Validation

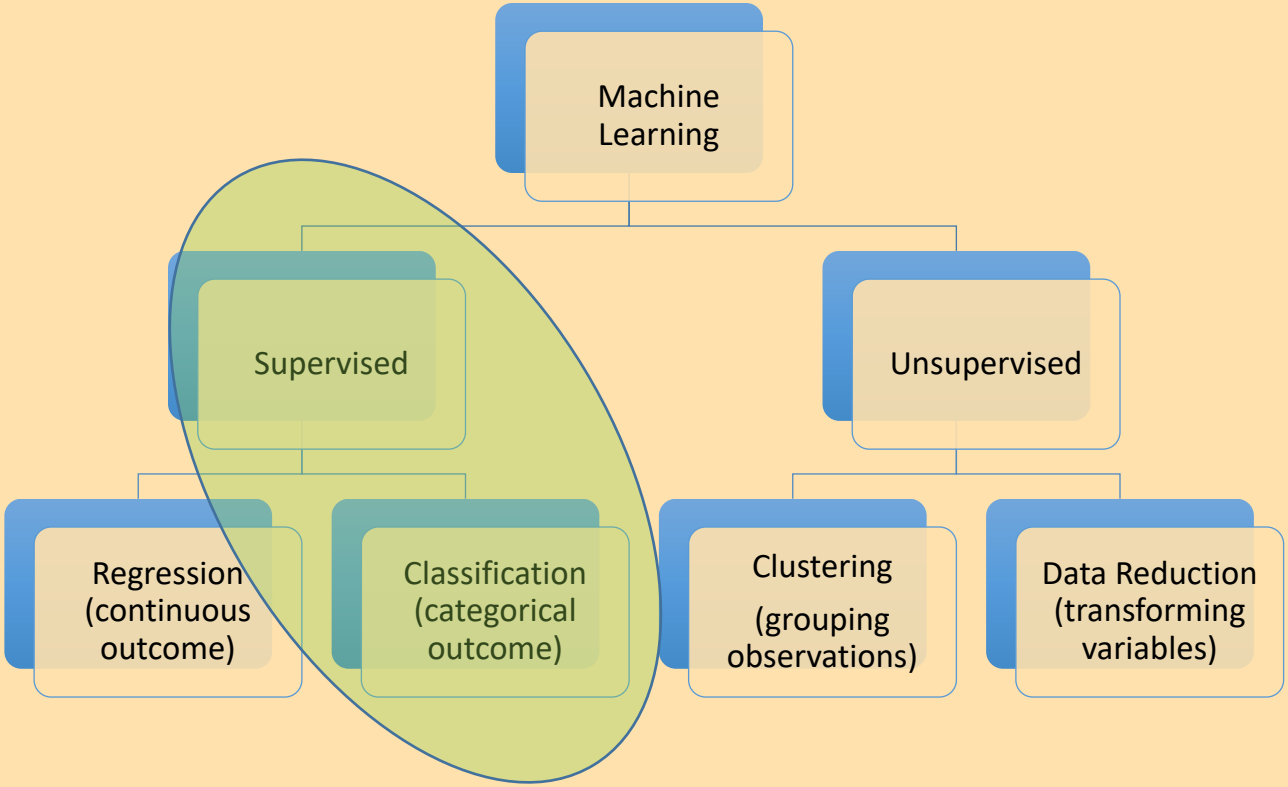


Overview of this lecture

- Introduction to classification
- Evaluation metrics for categorical data
- Validation and testing, continued
- Logistic regression

Classification

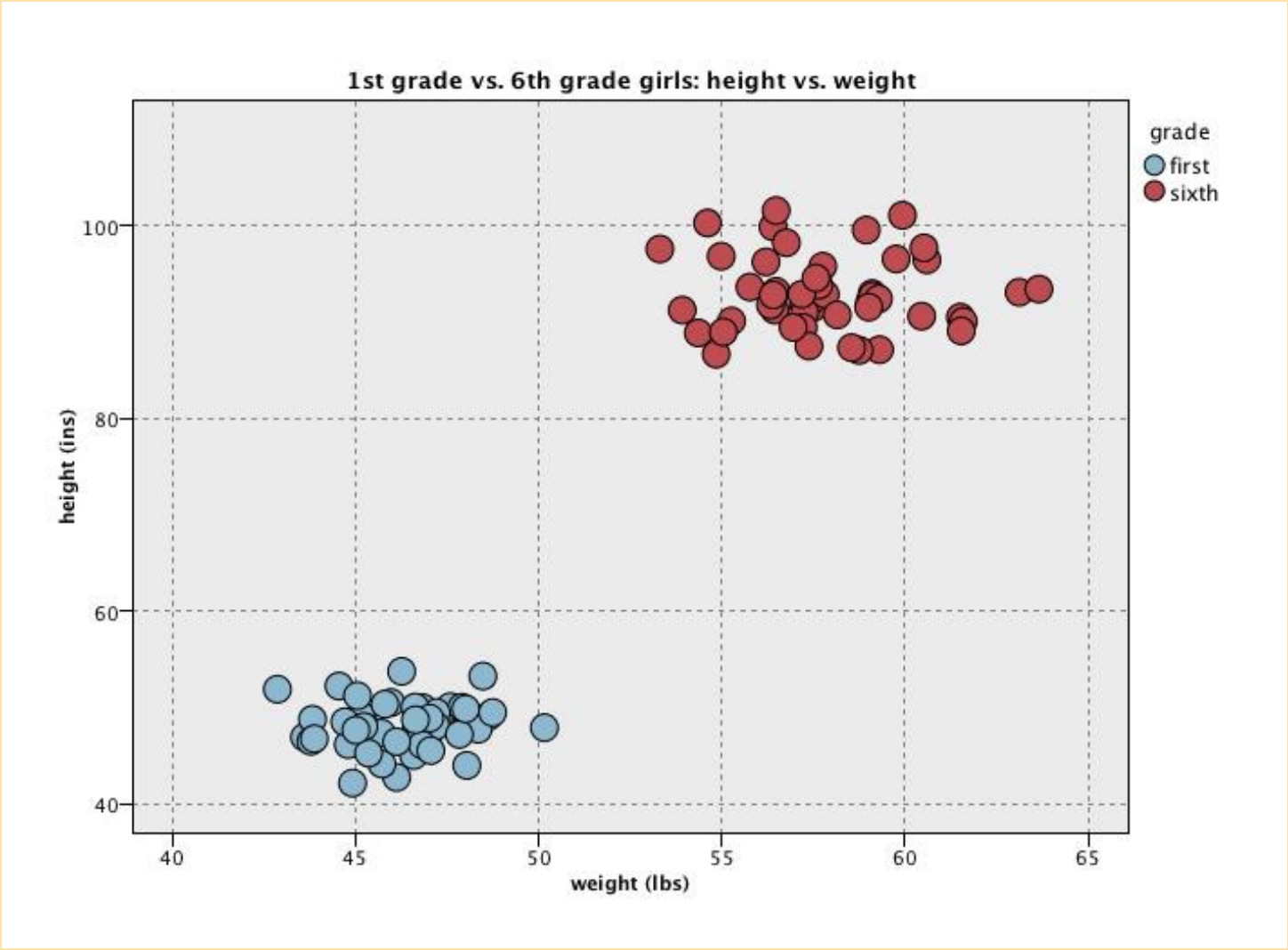
Classification



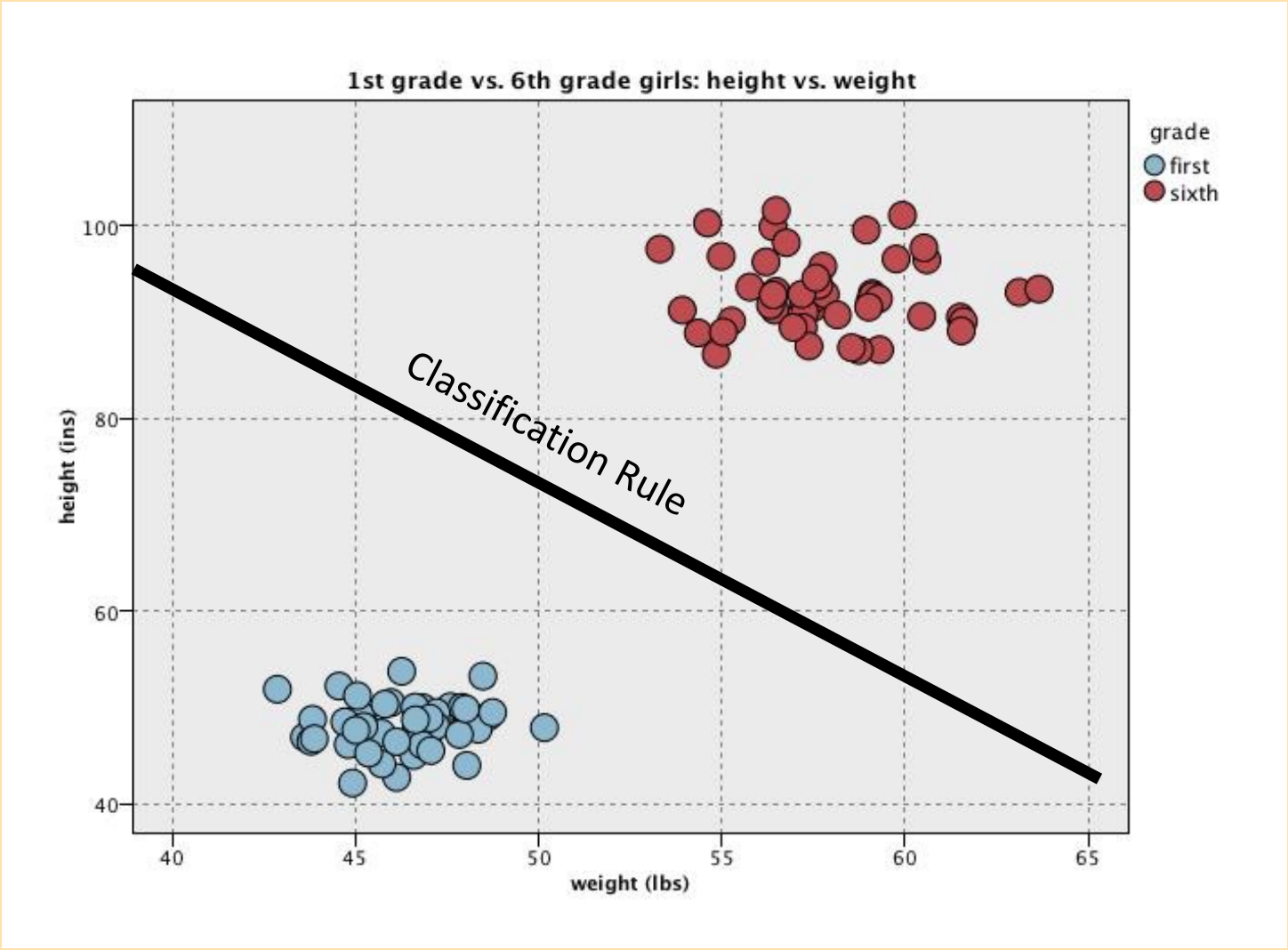
Classification

- Objective – given set of features, a categorical outcome, and some training data, generate a rule (classifier) to predict which category each instance belongs to
- Classifier is applied to new data where the outcome (i.e. target category) may or may not be known
- Let's look at a toy example trying to classify 1st vs. 6th grade girls based on height and weight….

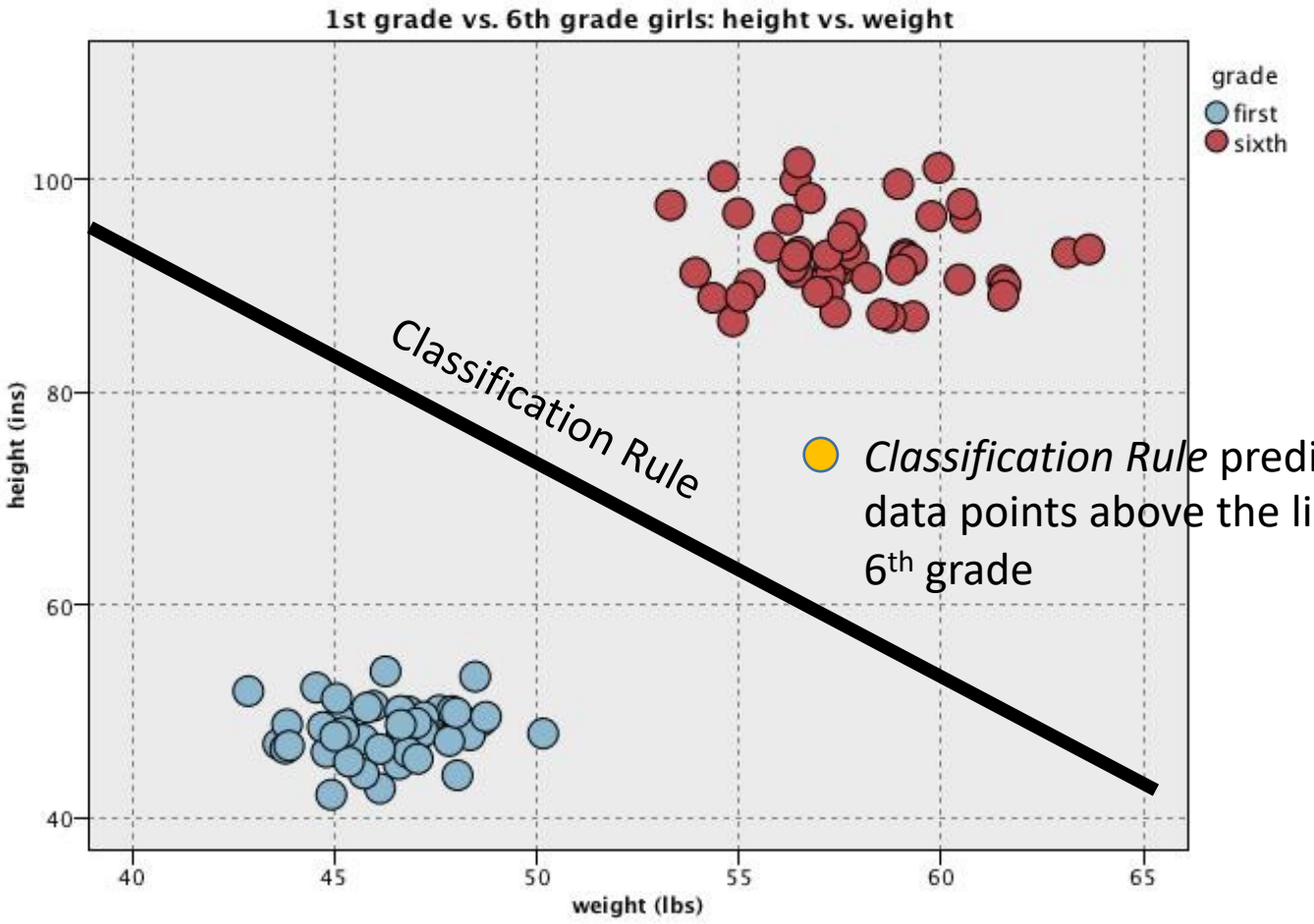
2D classification example



2D classification example

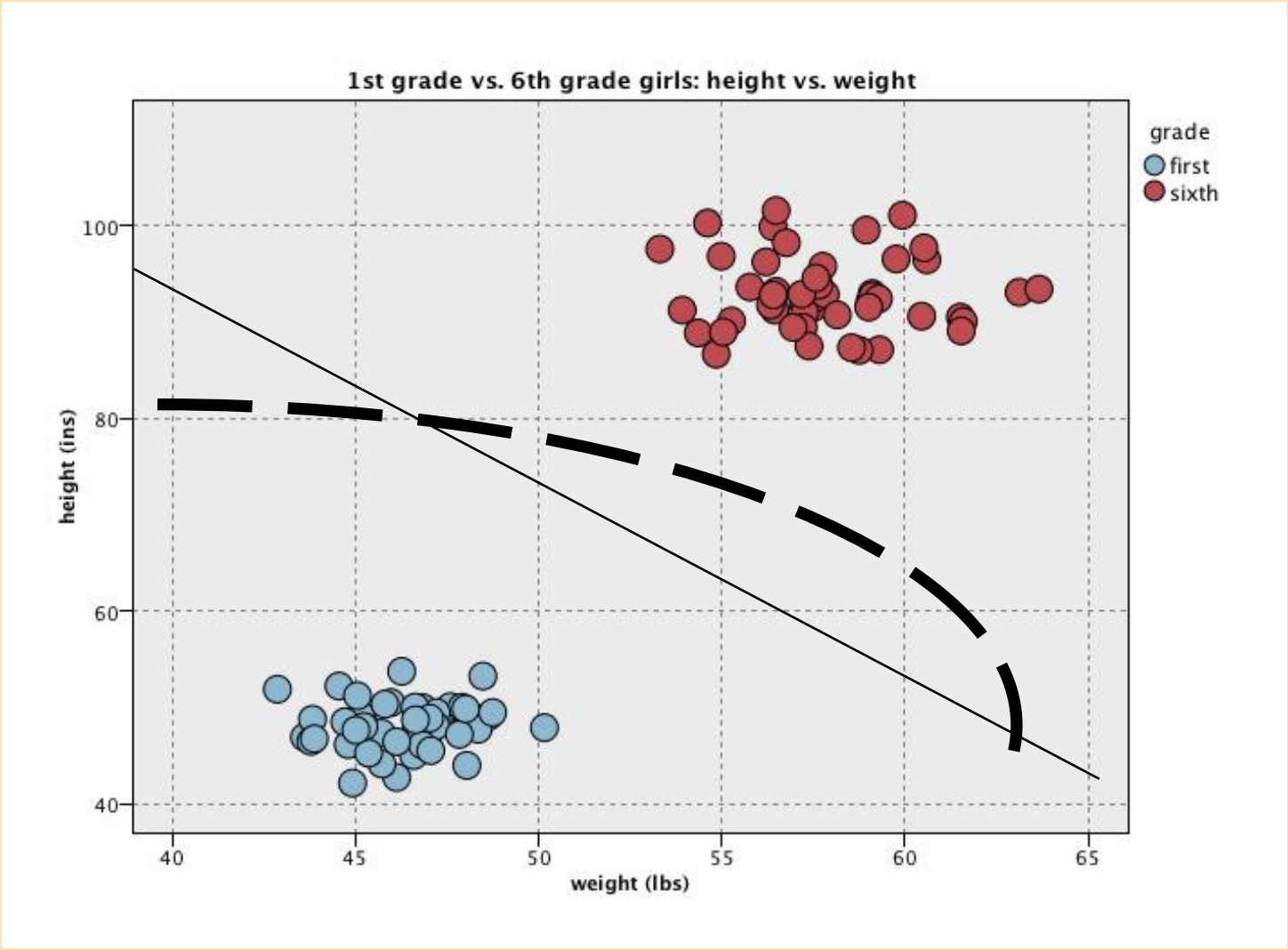


2D classification example



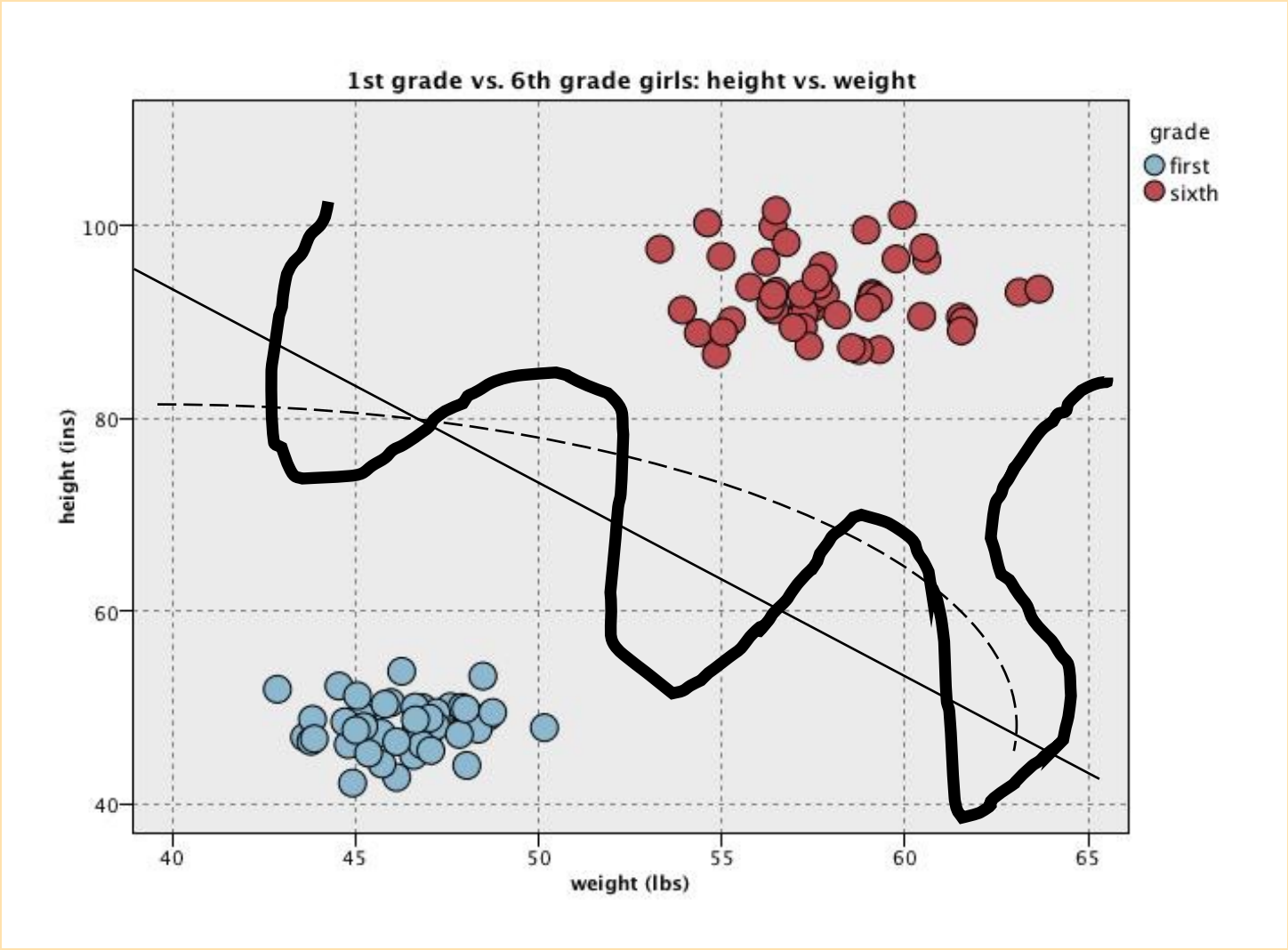
- *Classification Rule* predicts new data points above the line to be 6th grade

2D classification example



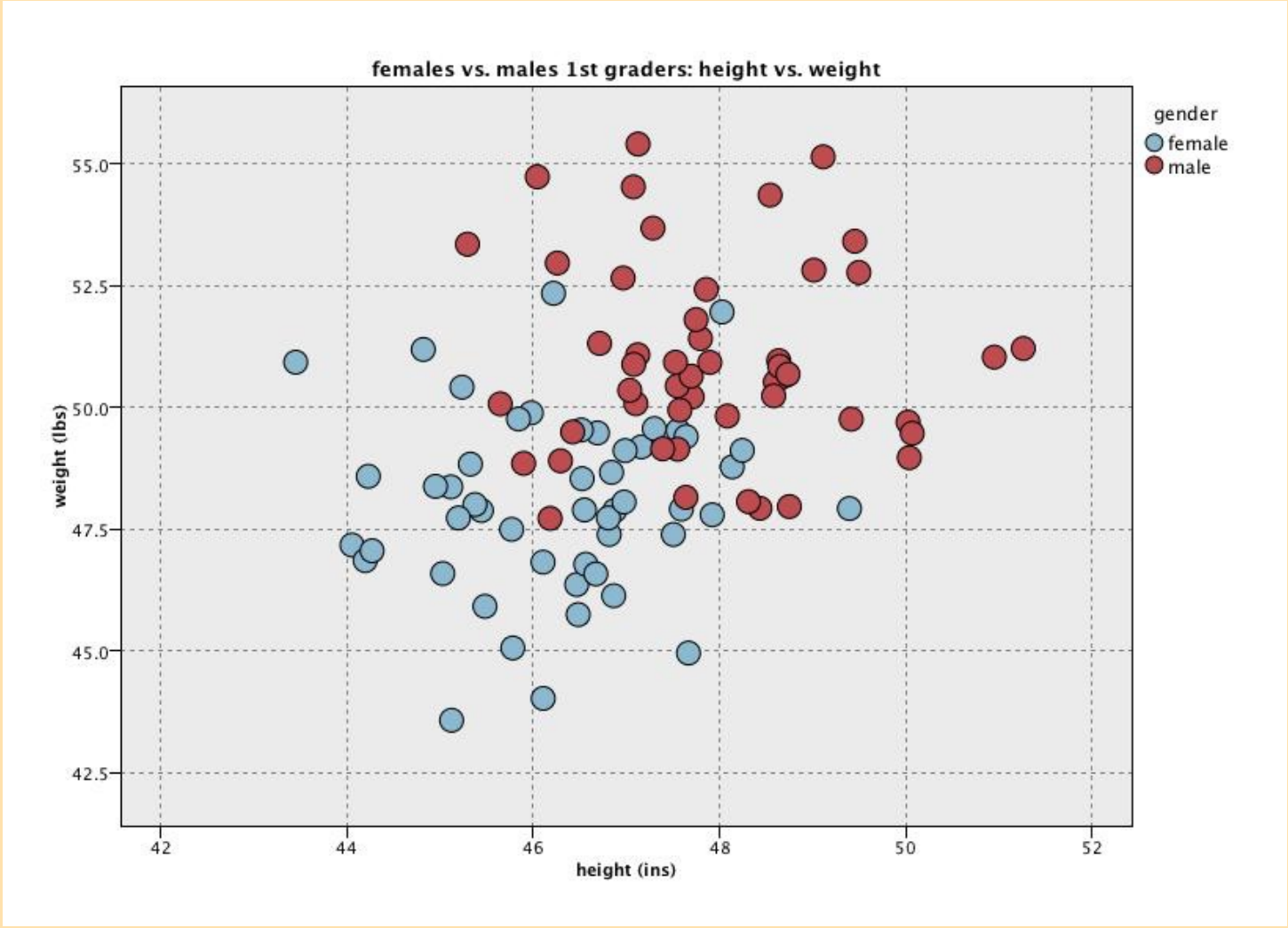
2D classification example

Many classification rules are possible…

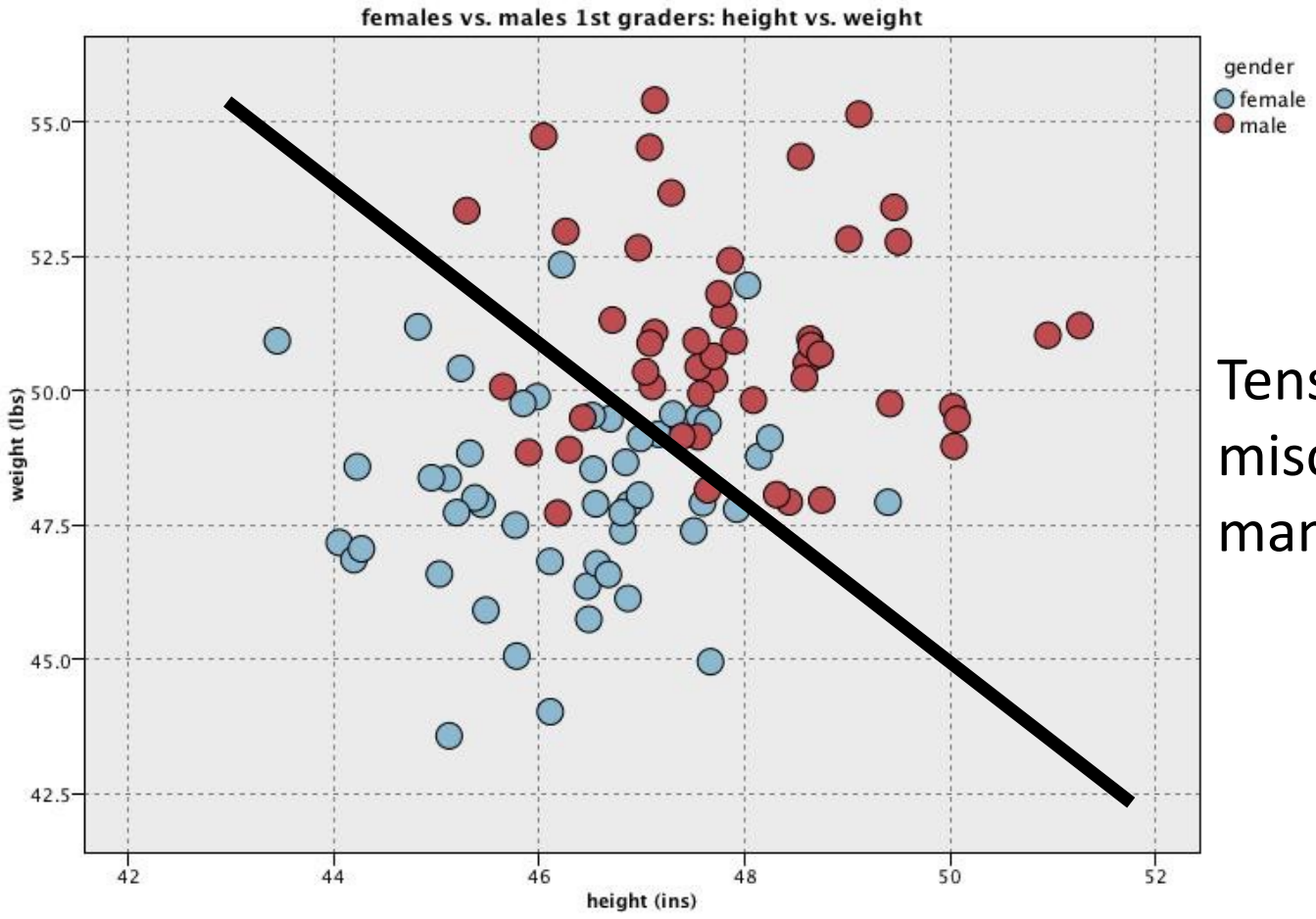


- But in real problems the separation is unlikely to be so neat
- Consider 1st grade boys vs. 1st grade girls

2D classification example

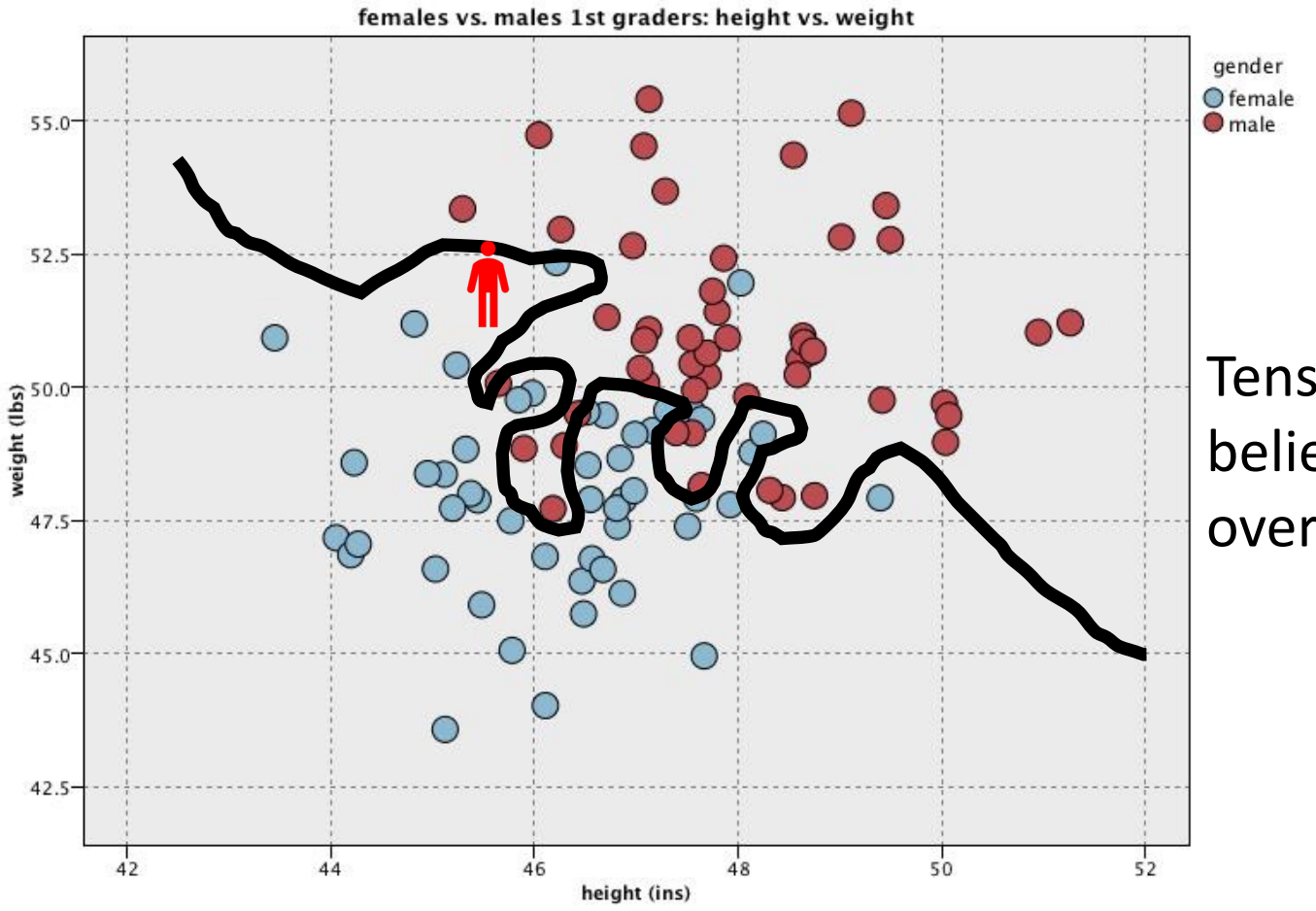


2D classification example



Tension: misclassifying many points

2D classification example

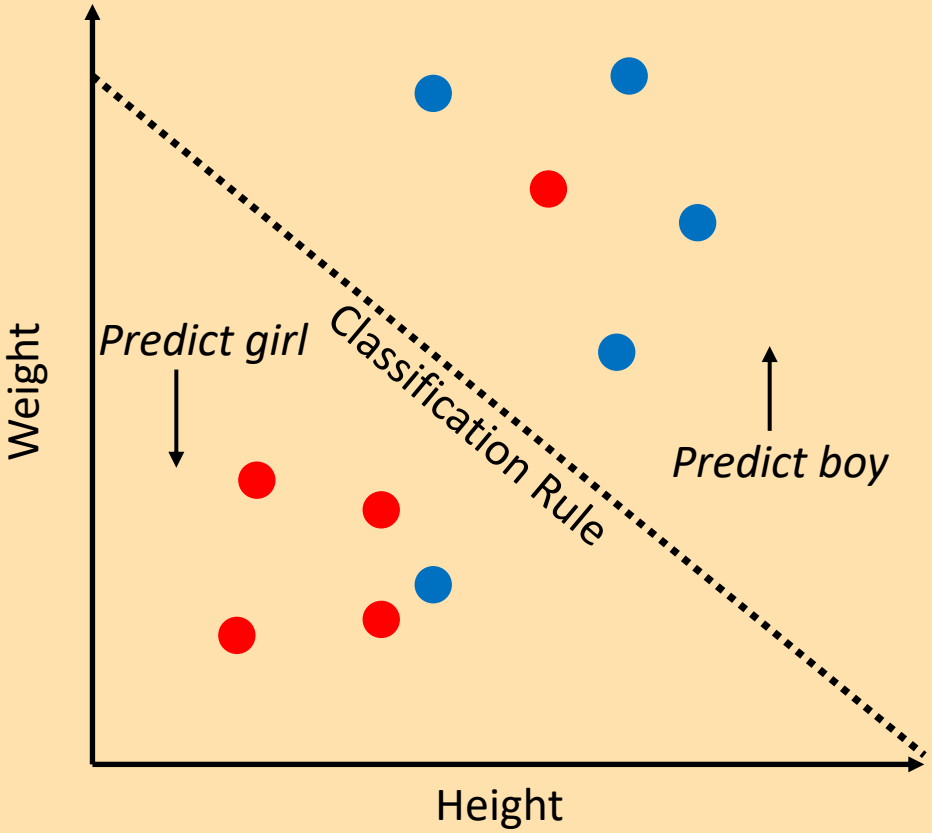


Tension: don't believe real - overfitting

The central tension in machine learning

- Want to find true predictive patterns in the data
- But, want to avoid overfitting spurious detail: i.e. don't want to make the model capture apparent patterns due to noise - we want results that are likely to be *generalizable* to future observations
- Solution – validation – reserve some data that has not been used in model fitting for the purpose of model evaluation and comparison
- To perform model evaluation for categorical outcomes, we'll need some definitions…

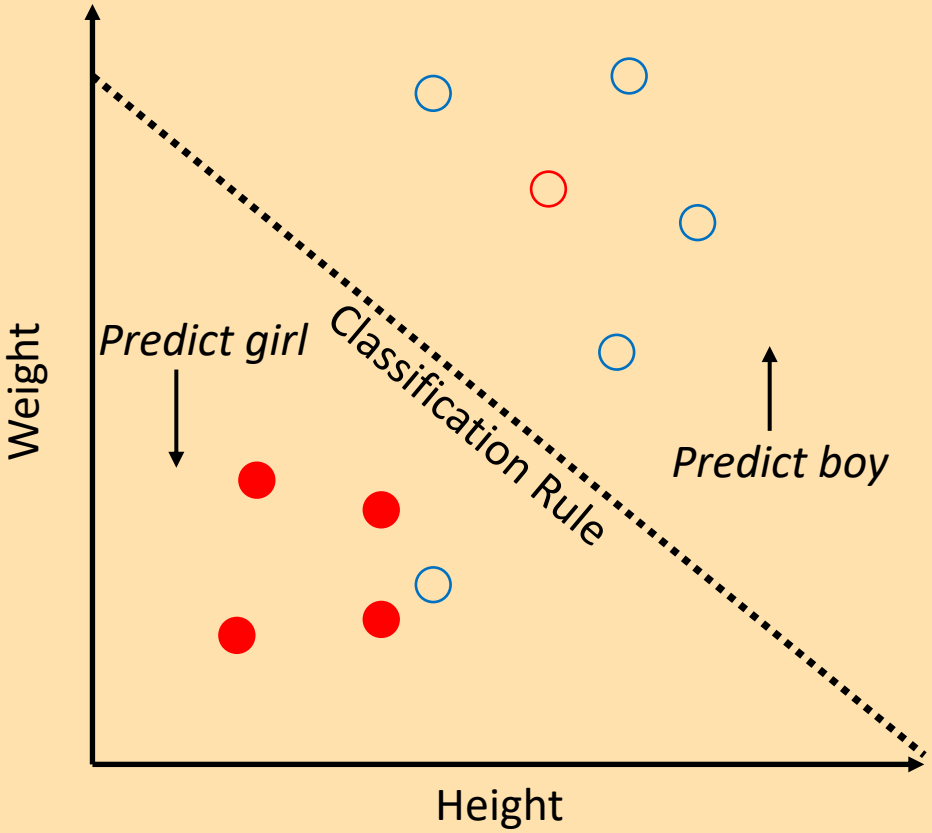
Definitions



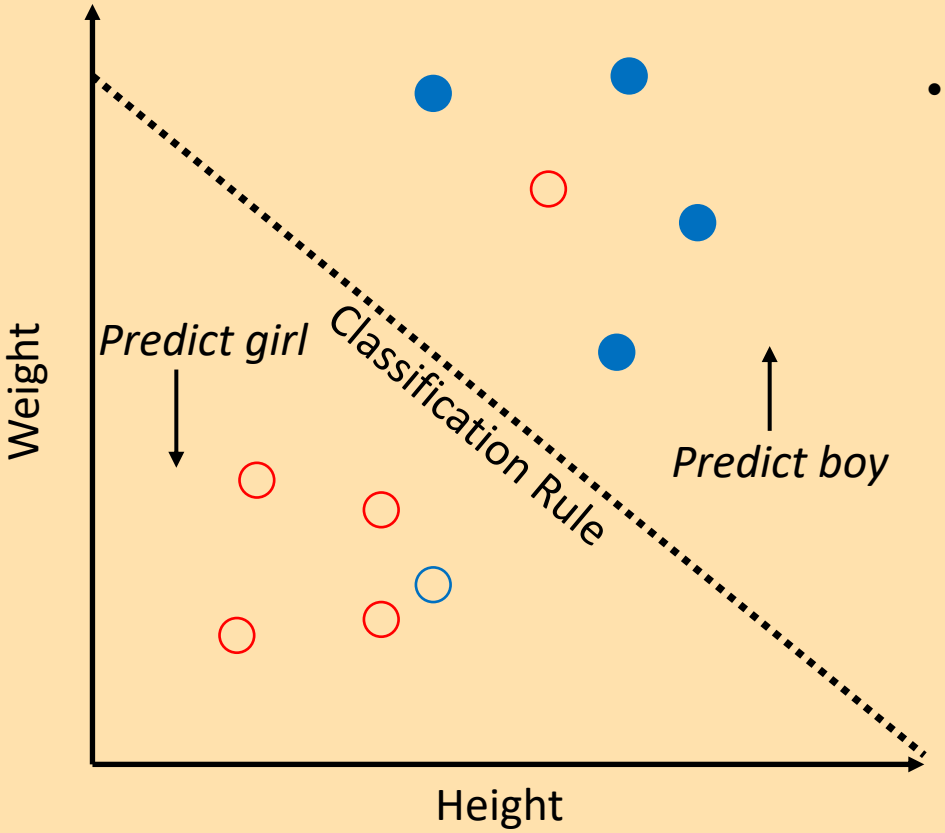
Girl (positive class) Boy (negative class)

Definitions

- True Positive (TP) – observation is positive and was predicted as positive



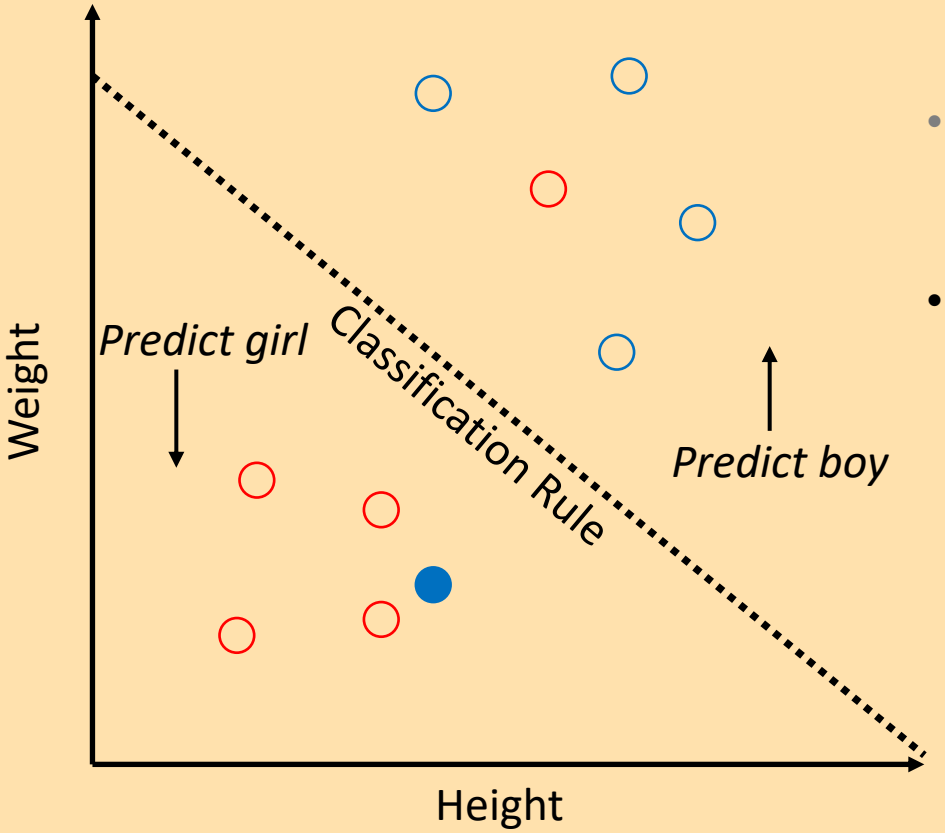
Definitions



- True Positive (TP) – observation is positive and was predicted as positive
- True Negative (TN) – observation is negative and was predicted as negative

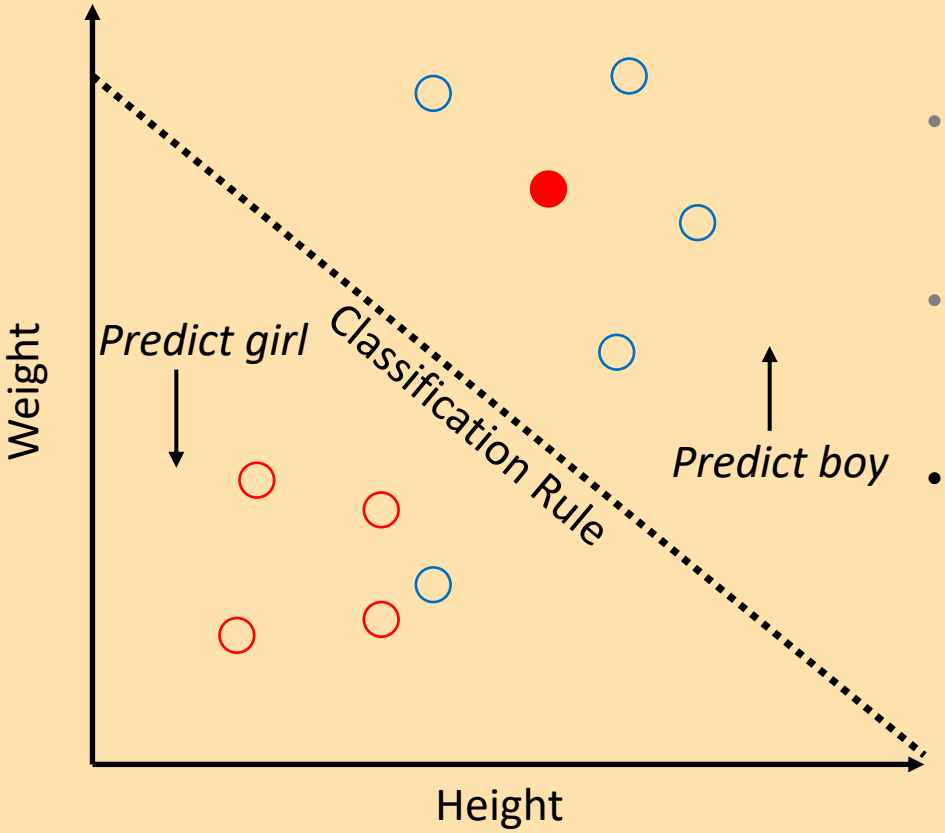
Girl (positive class) Boy (negative class)

Definitions



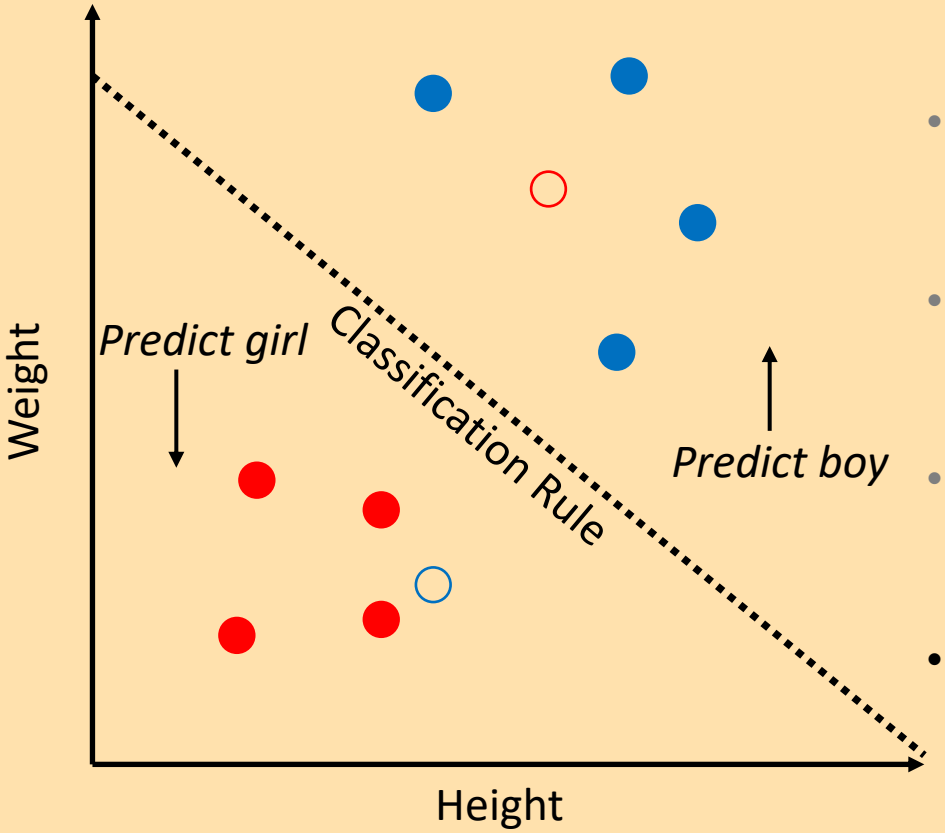
- True Positive (TP) – observation is positive and was predicted as positive
- True Negative (TN) – observation is negative and was predicted as negative
- False Positive (FP) – observation is negative and was predicted as positive

Definitions



- True Positive (TP) – observation is positive and was predicted as positive
- True Negative (TN) – observation is negative and was predicted as negative
- False Positive (FP) – observation is negative and was predicted as positive
- False Negative (FN) – observation is positive and was predicted as negative

Definitions



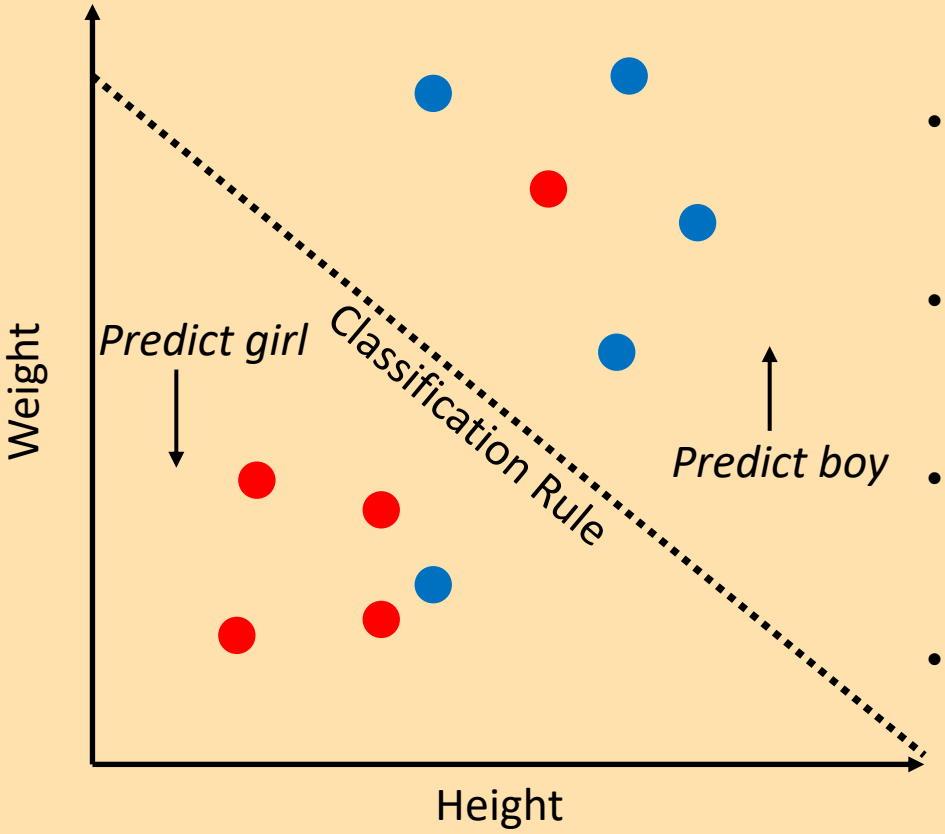
Girl (positive class) Boy (negative class)

- True Positive (TP) – observation is positive and was predicted as positive
- True Negative (TN) – observation is negative and was predicted as negative
- False Positive (FP) – observation is negative and was predicted as positive
- False Negative (FN) – observation is positive and was predicted as negative
- Classification accuracy – proportion of observations classified correctly

  = (#TP + #TN)/N

  = (4 + 4)/10 = 80%

Definitions



Girl (positive class) Boy (negative class)

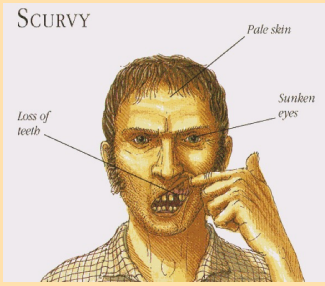
- True Positive (TP) – observation is positive and was predicted as positive
- True Negative (TN) – observation is negative and was predicted as negative
- False Positive (FP) – observation is negative and was predicted as positive
- False Negative (FN) – observation is positive and was predicted as negative
- Classification accuracy – proportion of observations classified correctly

= (#TP + #TN)/N

= (4 + 4)/10 = 80%

Cost-benefit

- Cost-benefit analysis – assessment of predictive performance when some errors matter mor than others
- Consider scurvy, a disease where treatment involves taking a simple cheap and benign medicine to cure it (e.g. taking vitamin C), but if the disease is undetected and thus untreated, the consequences are severe.



- In that case, I would want to put much more weight on missing disease (positives) than missing no disease (negatives); i.e. consequences of false negatives are much worse than false positives.
- For example, we might make the penalty (cost) for a false negative be 10 or 100 times that of a false positive

Confusion Matrix

Prediction\Truth	Negative	Positive	Total
Negative	#TN	#FN	#FN + #TN
Positive	#FP	#TP	#TP + #FP
Total	#FP + #TN	#TP + FN	N

Metrics obtained from confusion matrix:

$$\text{Accuracy} = \frac{\#\text{TP} + \#\text{TN}}{\text{N}} \qquad \text{Sensitivity} = \frac{\#\text{TP}}{\#\text{TP} + \#\text{FN}} \qquad \text{Specificity} = \frac{\#\text{TN}}{\#\text{TN} + \#\text{FP}}$$

Sensitivity (recall), Specificity, and Precision

- Sensitivity (recall) $= \frac{\#\mathrm{TP}}{\#\mathrm{TP}+\#\mathrm{FN}} =$
  Proportion of positives correctly identified as positive

- Specificity $= \frac{\#\mathrm{TN}}{\#\mathrm{TN}+\#\mathrm{FP}} =$
  Proportion of negatives correctly identified as negative

- Precision (positive predictive value) $= \frac{\#\mathrm{TP}}{\#\mathrm{TP}+\#\mathrm{FP}} =$
  Proportion of those identified positive that are positive

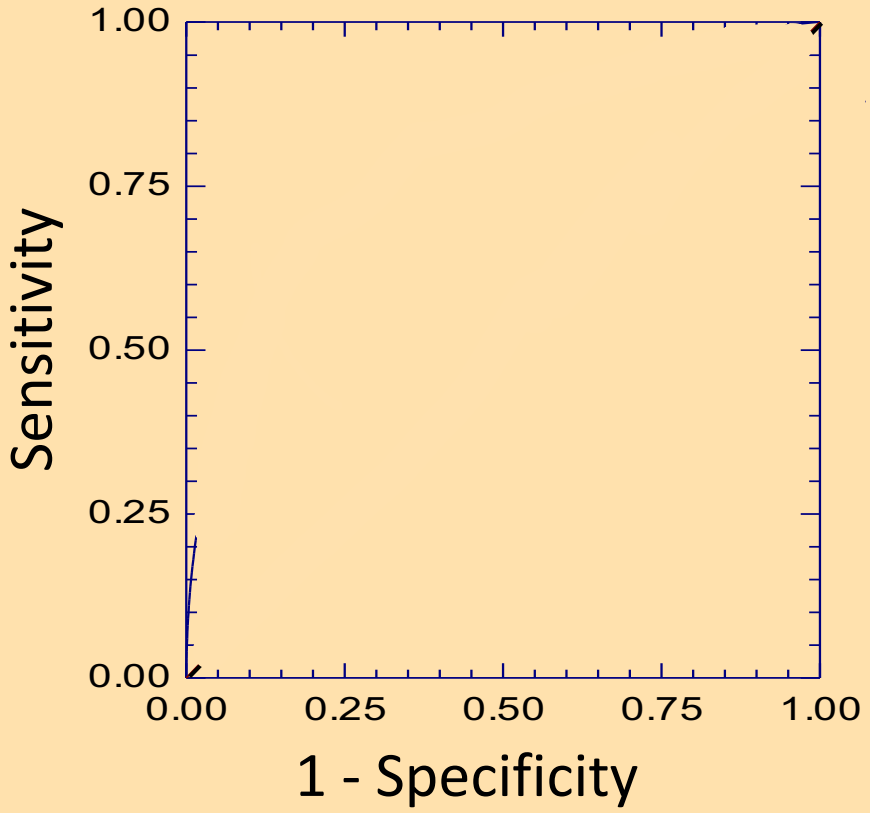
Sensitivity (recall), Specificity, and Precision: scurvy example

- The test has a 70% **sensitivity** if 70% of sailor scurvy+ are identified as scurvy+
- The test has a 40% **specificity** if 40% of sailor scurvy- are identified as scury-
- The test has a **precision** of 80% if 80% of sailors predicted to be scurvy+ are truly scuvry+

Sensitivity and Specificity: a trade-off

- There is often a trade-off here between sensitivity and specificity
- Imagine a classification procedure that gives you a probability of being positive for a disease (such classifiers are common)
- To classify individuals we need to choose a threshold to identify them as positive vs. negative (e.g. probability of disease > .70)
- Increasing the threshold increases specificity but reduces sensitivity
- Decreasing the threshold increases sensitivity but reduces specificity

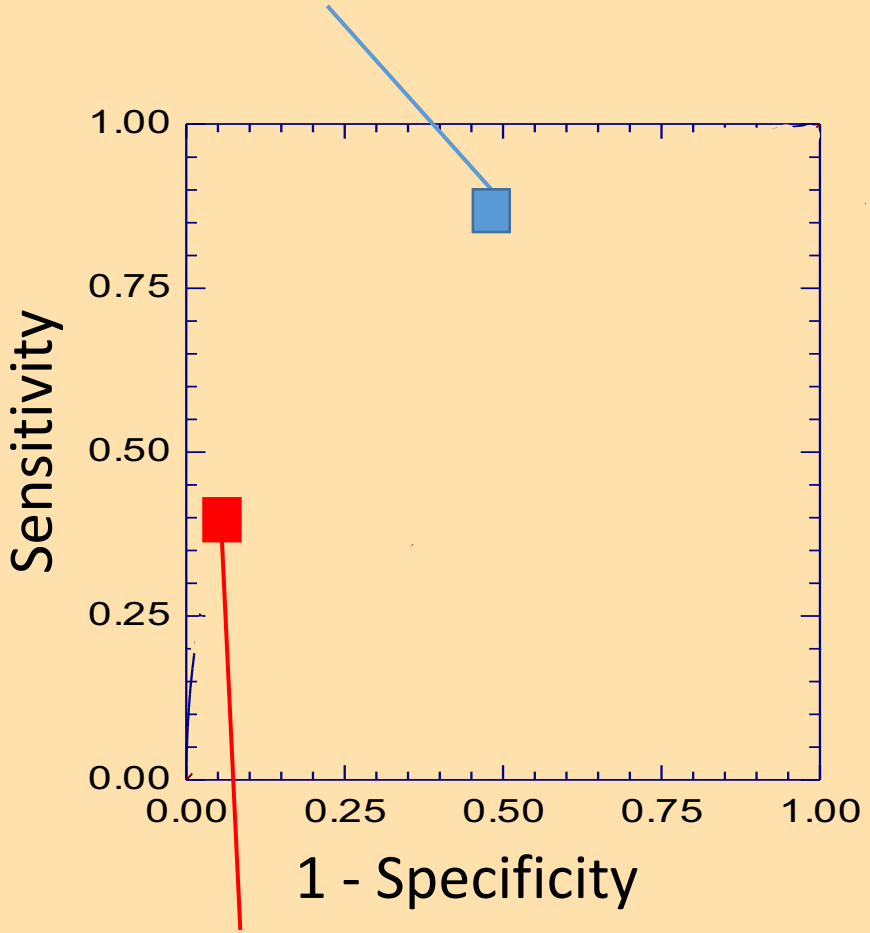
Receiver operating characteristic (ROC) curve



Definition: The ROC curve is a graphical summary of the performance of a classifier when the classification threshold is varied.

Choosing threshold based on ROC

Rule 1: predict label disease if probability of disease > .4

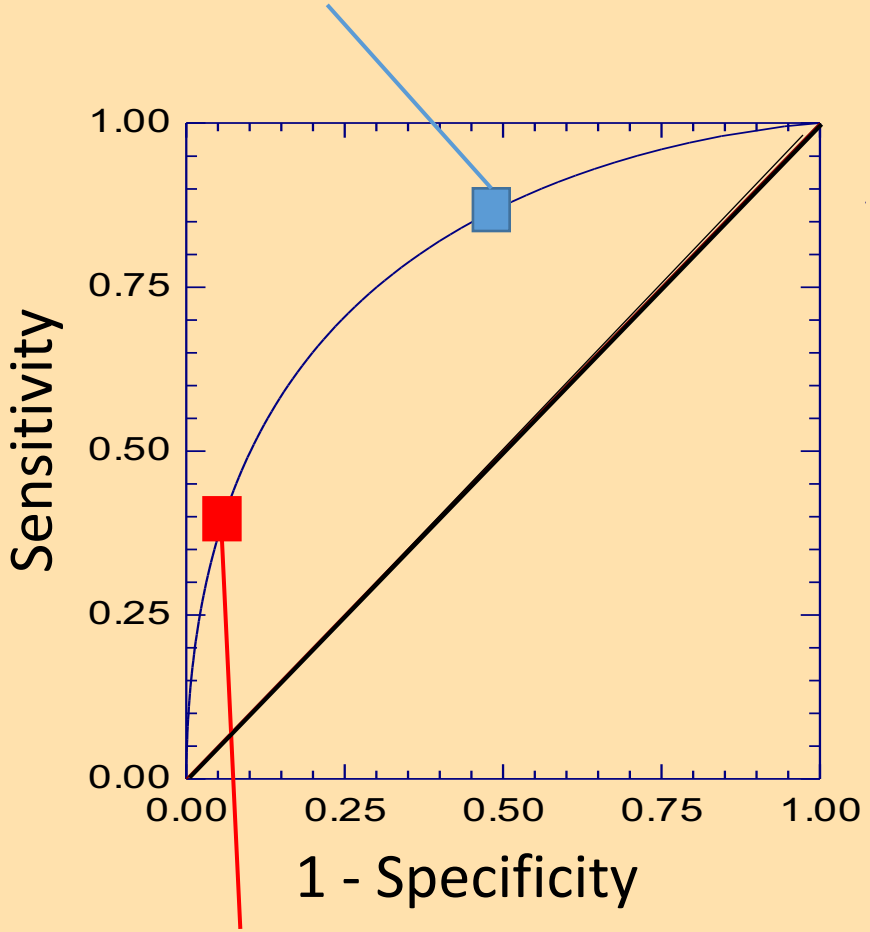


- Many possible thresholds, each corresponding to one point on curve
  - Rule 1: high sensitivity (80%) low specificity (50%)
  - Rule 2: low sensitivity (40%) high specificity (90%)
- ROC curve is constructed by evaluating each possible threshold

Rule 2: predict label disease if probability of disease > .80

Choosing threshold based on ROC

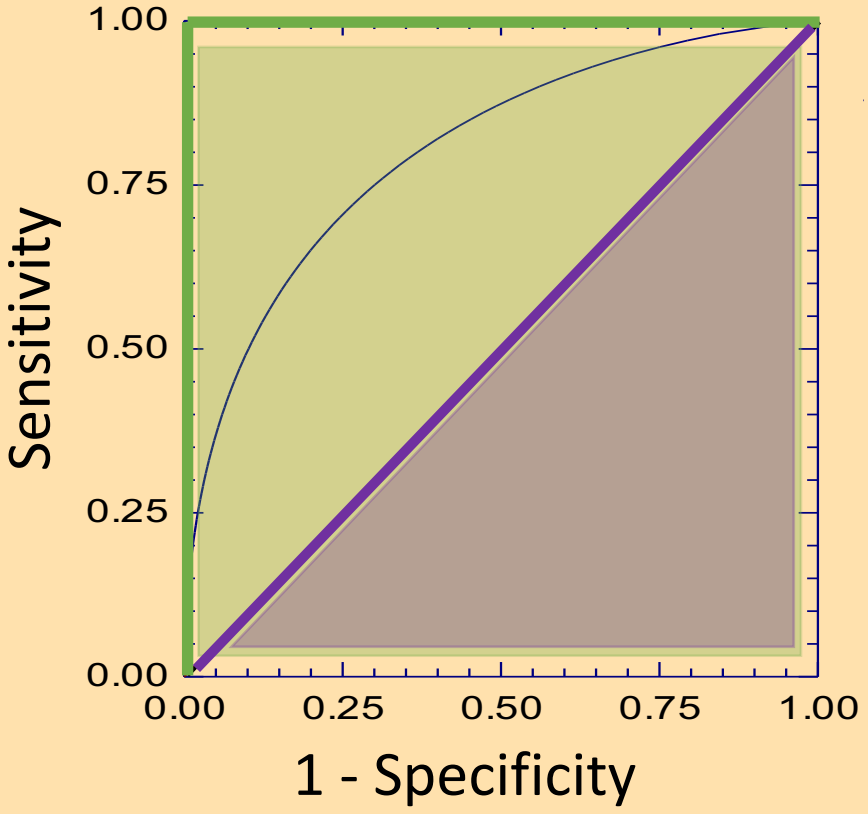
Rule 1: predict label disease if probability of disease > .4



- Many possible thresholds, each corresponding to one point on curve
  - Rule 1: high sensitivity (80%) low specificity (50%)
  - Rule 2: low sensitivity (40%) high specificity (90%)
- ROC curve is constructed by evaluating each possible threshold
- The black diagonal line represents performance from 50/50 guessing
- Choosing a single threshold depends on context, i.e. cost-benefit

Rule 2: predict label disease if probability of disease > .80

Area under the (ROC) curve (AUC)



- We will not be concerned with choosing thresholds, rather, we will use ROC to generate another evaluation metric
- AUC gives a measure of how well the predictor does across all possible thresholds
- Ideal AUC is **1**
- AUC for random predictor = **0.5**
- Objective is to find classifier that maximizes AUC
- In a sense, AUC tells you how well the classifier separates the two classes

AUC “Grade” for Classifier

AUC	Grade
.90 – 1.00	A
.80 - .89	B
.70 - .79	C
.60 - .69	D

Accuracy vs. AUC

- When should we choose accuracy and when AUC? Consider both the sample and the class imbalance.
- Accuracy (and cost-benefit) work well when your dataset is representative of the larger population (metric incorporates prevalence – e.g. proportion of people in population that have disease)
- AUC works better when your dataset is not representative of larger population, e.g. in “case-control” studies (metric is independent of prevalence)
- Also think about imbalance in class labels

Evaluation metrics for categorical data

- Possible evaluation metrics:
  - *Classification Accuracy* – or conversely, the *classification error rate* = 1 – classification accuracy
  - *Cost-Benefit* – some errors (e.g. FP, FN) are counted more than other, often context dependent
  - Sensitivity, specificity, precision
  - Area Under Receiver Operating Characteristic (ROC) Curve (AUC)
- Guiding questions in selecting evaluation metrics
  - What is my analytical goal, i.e. research question?
  - What metric can best help me evaluate how reliable by predicted results are in the context of my analytical goal?

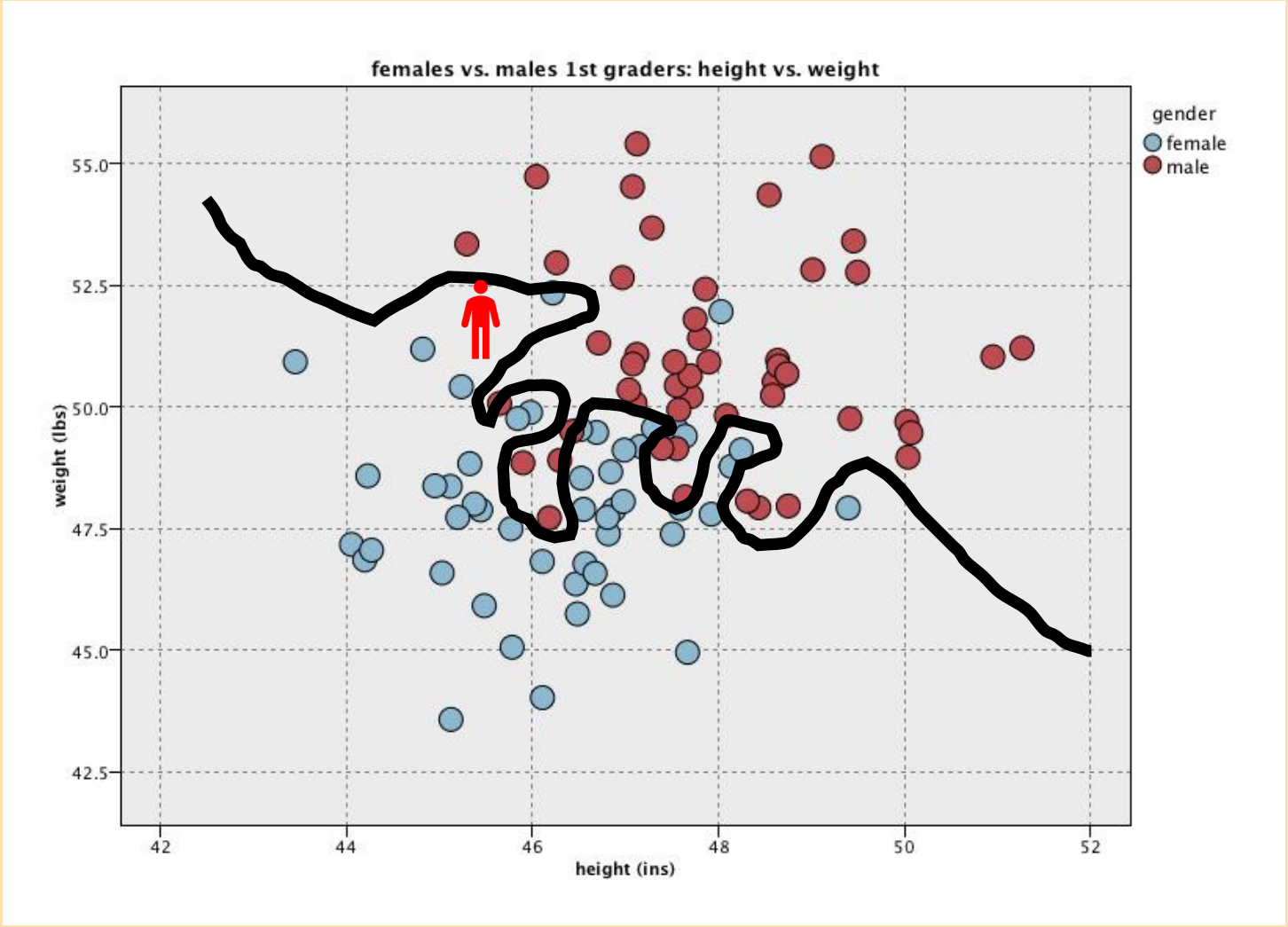
Training set error rate: **is generally too optimistic!**

You can find patterns even in random data

So we need to evaluate on separate independent (*test*) set…

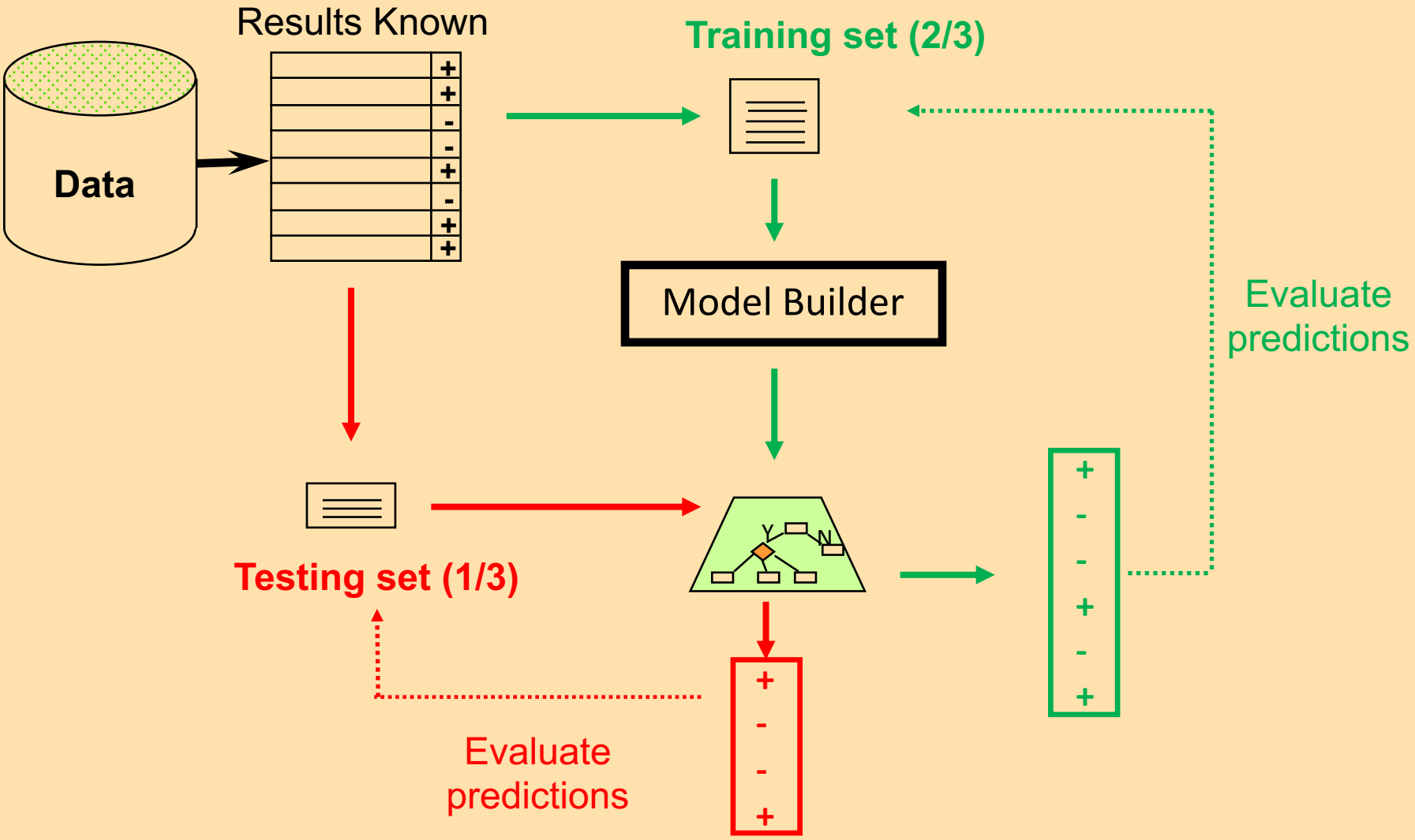
and now we have evaluation metrics to evaluate our classifications!

2D Classification example



At least greater than a few hundred instances and better >1000

Training/Test



Classification methods

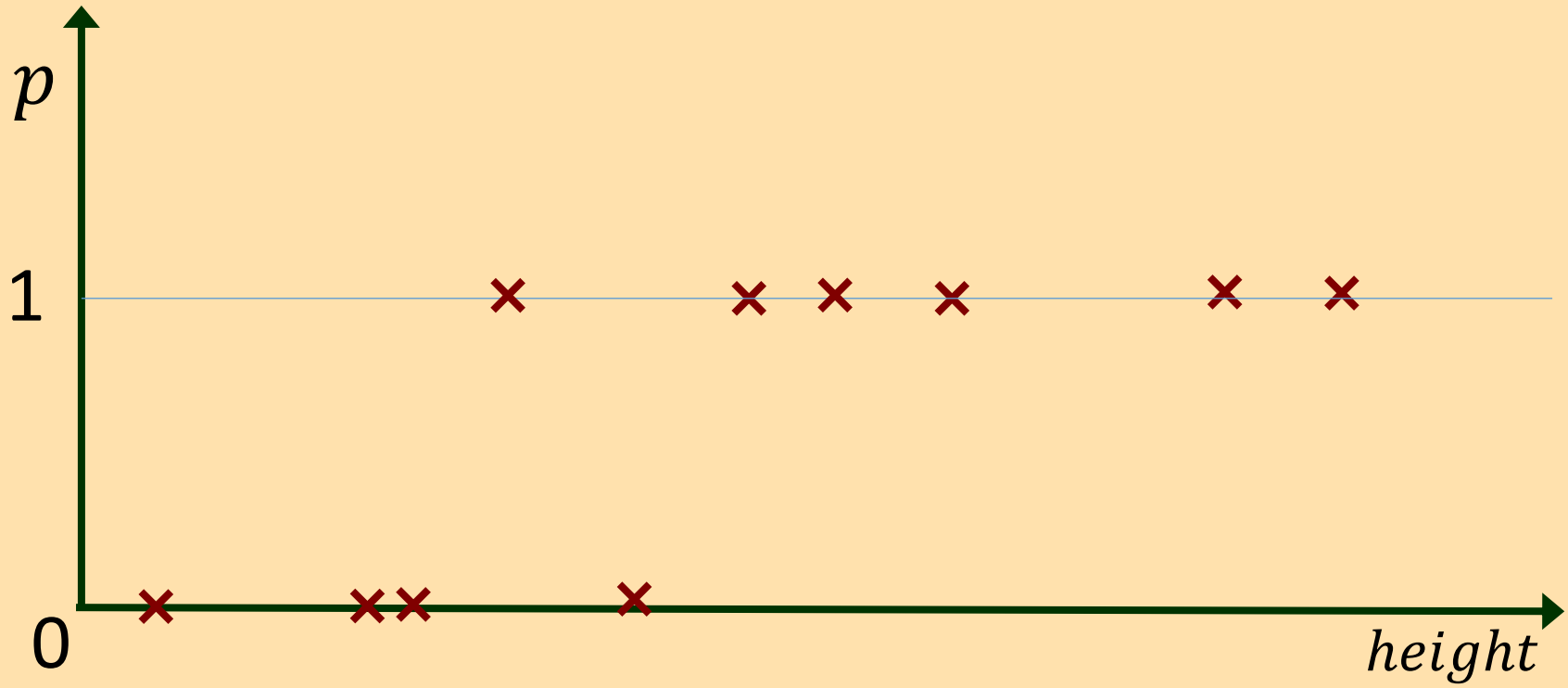
- Logistic regression
- K-nearest neighbors
- Classification trees
- Support vector machines (SVM)
- Neural networks

Logistic Regression

Thought experiment

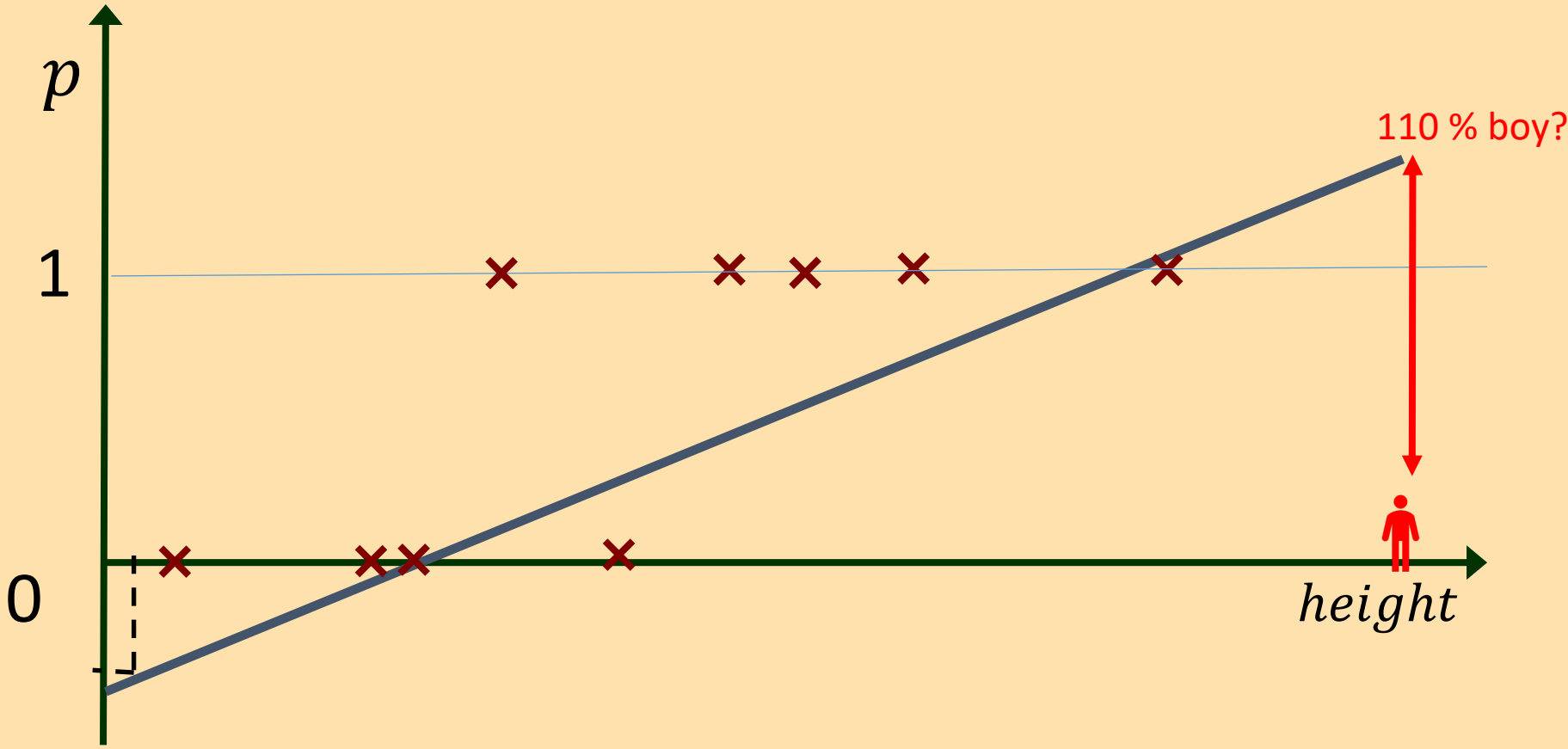
- We want a model where the probability of being in a class is based on the values of predictor(s)
- What about using linear regression?
- $p = a * height + c$, where p is the probability of being in class 1 vs. 0, e.g. $p =$ probability boy (=1) vs. girl (=0)
- What might be the concern here?

Linear regression?



Linear regression

Problem predictions – outside range 0 to 1



Class outcome regression modeling

- We want a model where the probability of being in a class depends on predictor(s)
- What about using linear regression?
- $p = a * height + c$, where p is the probability of being in class 1 vs. 0, e.g. $p =$ probability boy (=1) vs. girl (=0)
- What might be the concern here?
- Possible values for p are not restricted to be between 0 and 1 (as required for probabilities) – can be >1 or <0

Logistic regression

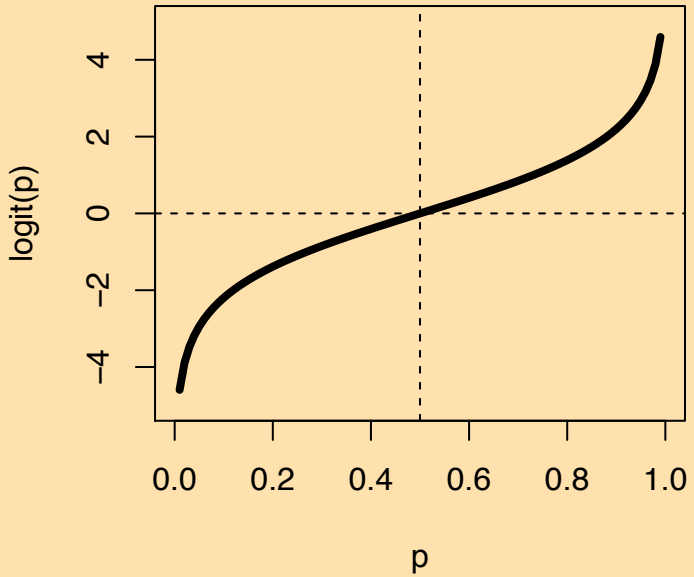
p must be between 0 and 1 (probability)

We can instead take the *odds* which is between 0 and infinity: $\frac{p}{1-p} > 0$ but still not quite what we need…

to get values in the range 0 to 1, we take the log of the odds (*logit function*) and get the logistic regression equation:

$$\text{logit}(p) = \log\left(\frac{p}{1-p}\right) = a * height + c$$

We can extend logistic regression to multiple predictors just as we did for linear regression.



Logistic regression for classification

- When fitting logistic regression we get a model that relates predictors to a category probability
- To convert the probability to a class we need to threshold the probability
- E.g. if probability $p \leq 0.5$ then class as girl, otherwise if $p > 0.5$ class as boy

Logistic regression

- A long history (relatively) – well known
- Much statistical theory
- Works well for many problems
- Can be extended to multiple categories
- Can provide interpretable models

Review of this lecture

- Introduction to classification
- Evaluation metrics for classes
- Validation and testing, continued
- Logistic regression

Next Monday – more classification!

- K-nearest neighbors
  - Parameter tuning
  - Training-validation-test
  - Cross-validation
- Classification trees (recursive partitioning)
- Support vector machines
- Artificial neural networks
- Comparison of classifiers