

# EVALUATION

David Kauchak  
CS 1.58 – Spring 2022

1

## Admin

### Assignment 3

- ClassifierTimer class

### Reading

2

## So far...

1. Throw out outlier examples
2. Remove noisy features
3. Pick "good" features
4. Normalize feature values
  1. center data
  2. scale data (either variance or absolute)
5. Normalize example length
6. **Finally, train your model!**

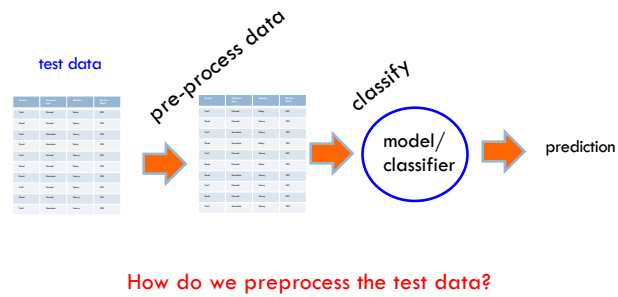
3

## What about testing?



4

## What about testing?



5

## Test data preprocessing

1. Throw out outlier examples
2. Remove noisy features
3. Pick "good" features
4. Normalize feature values
  1. center data
  2. scale data (either variance or absolute)
5. Normalize example length

Which of these do we need to do on test data?  
Any issues?

6

## Test data preprocessing

- ~~1. Throw out outlier examples~~
2. Remove irrelevant/noisy features Remove/pick same features
3. Pick "good" features Do these
4. Normalize feature values Do these
  1. center data Do this
  2. scale data (either variance or absolute) Do this
5. Normalize example length

Whatever you do on training, you have to do the EXACT same on testing!

7

## Normalizing test data

For each feature (over all examples):

**Center:** adjust the values so that the mean of that feature is 0: subtract the **mean** from all values

Rescale/adjust feature values to avoid magnitude bias:

- ▣ **Variance scaling:** divide each value by the **std dev**
- ▣ **Absolute scaling:** divide each value by the **largest value**

What values do we use when normalizing testing data?

8

## Normalizing test data

For each feature (over all examples):

**Center:** adjust the values so that the mean of that feature is 0: subtract the **mean** from all values

Rescale/adjust feature values to avoid magnitude bias:

- ▣ **Variance scaling:** divide each value by the **std dev**
- ▣ **Absolute scaling:** divide each value by the **largest value**

Save these from training normalization!

9

## Normalizing test data

training data  
(labeled examples)

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16
17	18	19	20
21	22	23	24
25	26	27	28
29	30	31	32
33	34	35	36
37	38	39	40
41	42	43	44
45	46	47	48
49	50	51	52
53	54	55	56
57	58	59	60
61	62	63	64
65	66	67	68
69	70	71	72
73	74	75	76
77	78	79	80
81	82	83	84
85	86	87	88
89	90	91	92
93	94	95	96
97	98	99	100



1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16
17	18	19	20
21	22	23	24
25	26	27	28
29	30	31	32
33	34	35	36
37	38	39	40
41	42	43	44
45	46	47	48
49	50	51	52
53	54	55	56
57	58	59	60
61	62	63	64
65	66	67	68
69	70	71	72
73	74	75	76
77	78	79	80
81	82	83	84
85	86	87	88
89	90	91	92
93	94	95	96
97	98	99	100

learn



test data

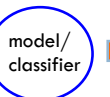
1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16
17	18	19	20
21	22	23	24
25	26	27	28
29	30	31	32
33	34	35	36
37	38	39	40
41	42	43	44
45	46	47	48
49	50	51	52
53	54	55	56
57	58	59	60
61	62	63	64
65	66	67	68
69	70	71	72
73	74	75	76
77	78	79	80
81	82	83	84
85	86	87	88
89	90	91	92
93	94	95	96
97	98	99	100

pre-process data

mean, std dev, max,...



classify



prediction

10

## Features pre-processing summary

Many techniques for preprocessing data

Which ones will work well will depend on the data and the classifier

Try them out and evaluate how they affect performance on dev data

Make sure to do the **exact same** pre-processing on train and test

1. Throw out outlier examples
2. Remove noisy features
3. Pick "good" features
4. Normalize feature values
  1. center data
  2. scale data (either variance or absolute)
5. Normalize example length

11

## Supervised evaluation

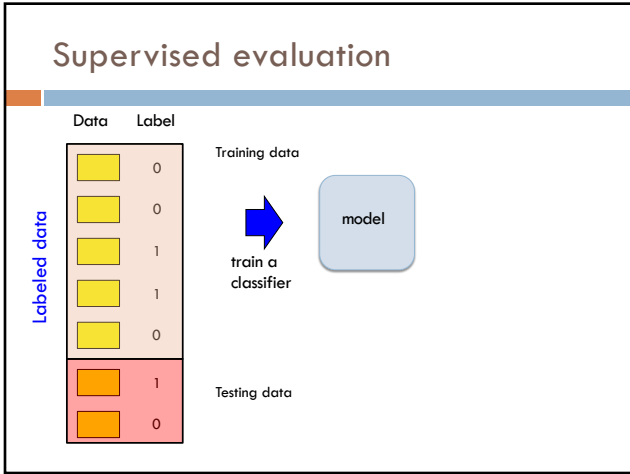
Data Label

Yellow	0
Yellow	0
Yellow	1
Yellow	1
Yellow	0
Orange	1
Orange	0

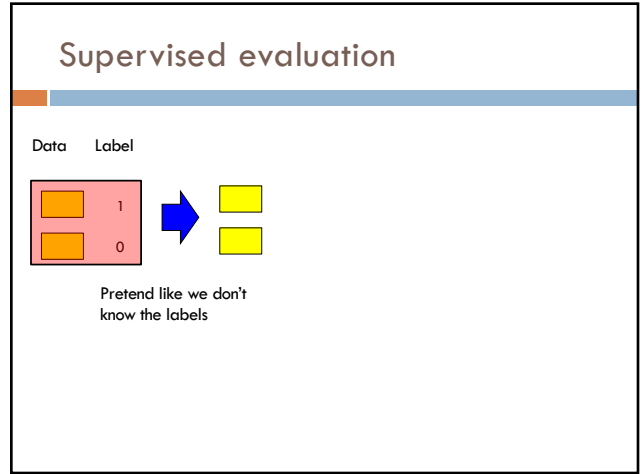
Training data

Testing data

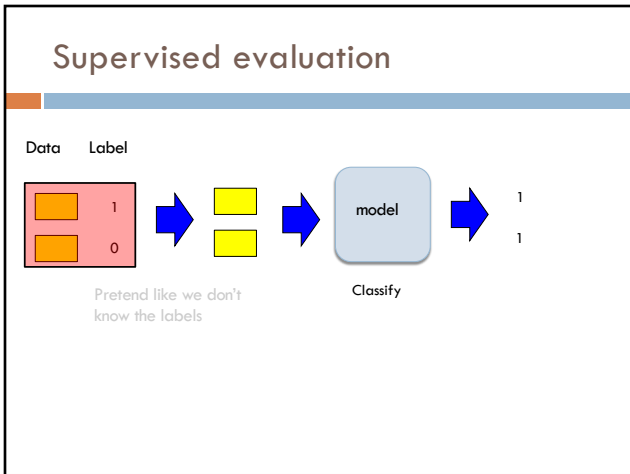
12



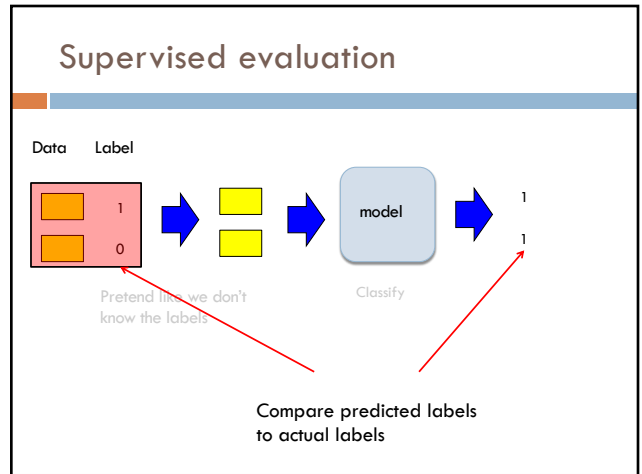
13



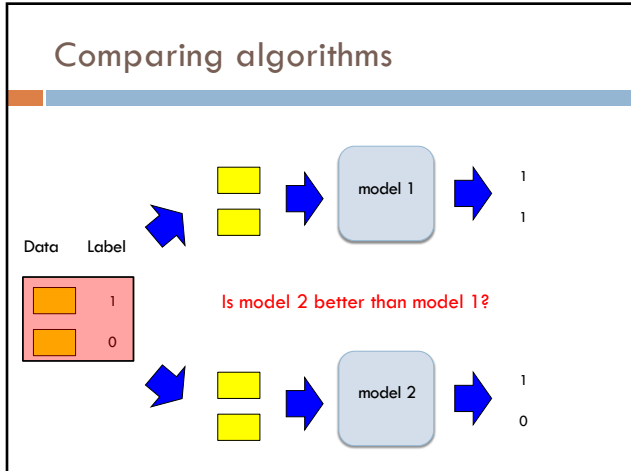
14



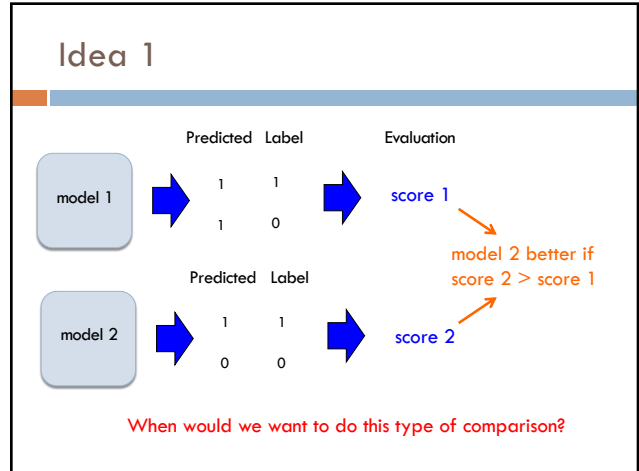
15



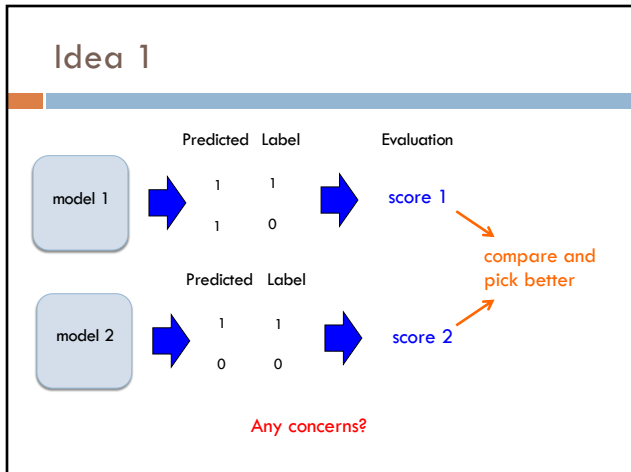
16



17



18



19

### Is model 2 better?

Model 1: 85% accuracy  
 Model 2: 80% accuracy

Model 1: 85.5% accuracy  
 Model 2: 85.0% accuracy

Model 1: 0% accuracy  
 Model 2: 100% accuracy

20

## Comparing scores: significance

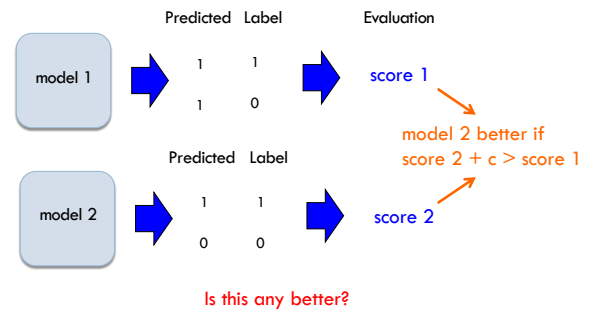
Just comparing scores on one data set isn't enough!

We don't just want to know which system is better on *this particular data*, we want to know if model 1 is better than model 2 *in general*

Put another way, we want to be confident that the difference is real and not just due to random chance

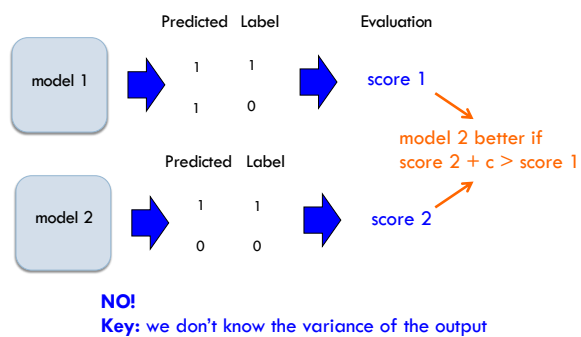
21

## Idea 2



22

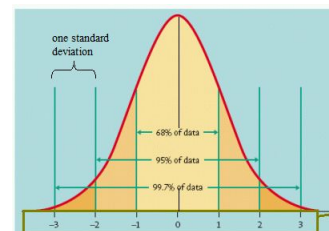
## Idea 2



23

## Variance

Recall that variance (or standard deviation) helped us predict how likely certain events are:

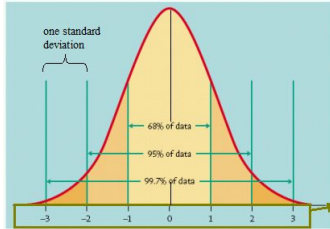


How do we know how variable a model's accuracy is?

24

## Variance

Recall that variance (or standard deviation) helped us predict how likely certain events are:



We need multiple accuracy scores! Ideas?

25

## Repeated experimentation

	Data	Label	
Labeled data	Yellow	0	Training data
	Yellow	0	
	Yellow	1	
	Yellow	1	
	Yellow	0	
	Orange	1	Testing data
	Orange	0	

Rather than just splitting once, split multiple times

26

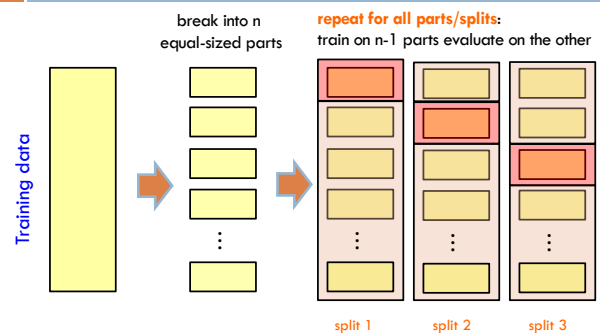
## Repeated experimentation

	Data	Label	Data	Label	Data	Label	
Training data	Yellow	0	Yellow	0	Orange	0	...
	Yellow	0	Orange	0	Yellow	0	
	Yellow	1	Yellow	1	Yellow	1	
	Yellow	1	Orange	1	Orange	1	
	Orange	0	Yellow	0	Yellow	0	
	Orange	1	Yellow	1	Yellow	1	

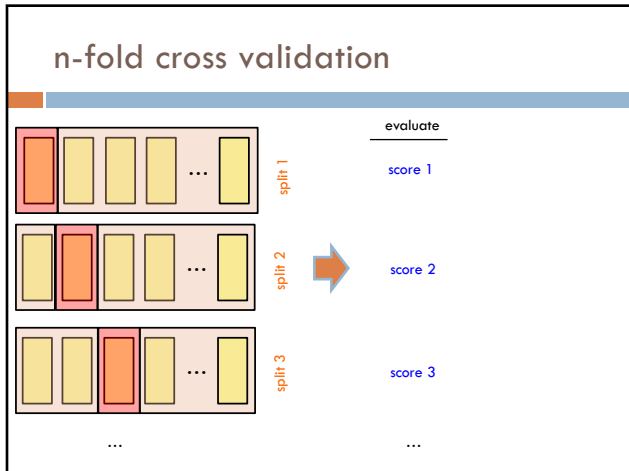
= train  
 = development

27

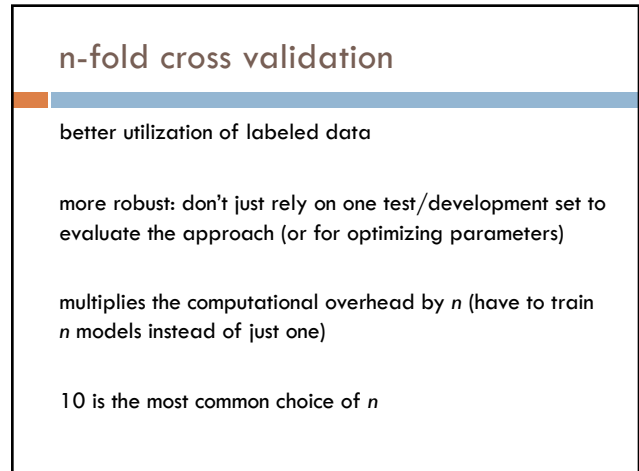
## n-fold cross validation



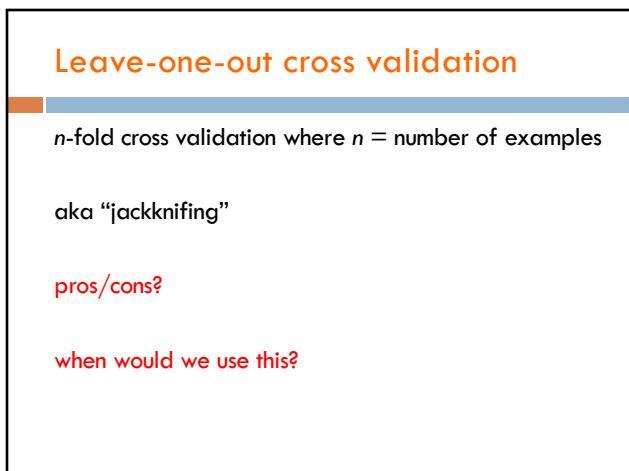
28



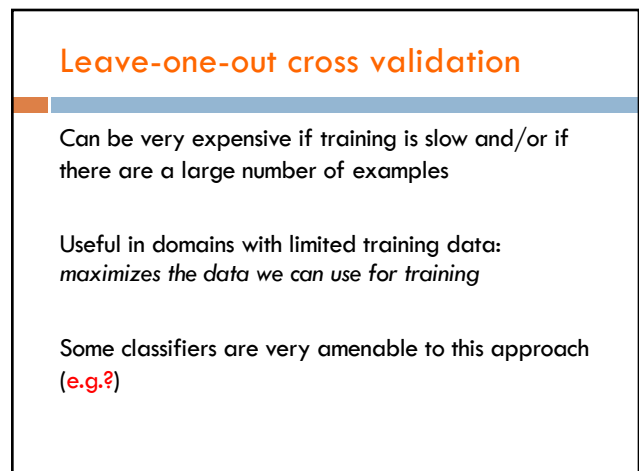
29



30



31



32



## Comparing systems: sample 1

split	model 1	model 2
1	87	88
2	85	84
3	83	84
4	80	79
5	88	89
6	85	85
7	83	81
8	87	86
9	88	89
10	84	85
<b>average:</b>	<b>85</b>	<b>85</b>

Is model 2 better than model 1?

33

## Comparing systems: sample 2

split	model 1	model 2
1	87	87
2	92	88
3	74	79
4	75	86
5	82	84
6	79	87
7	83	81
8	83	92
9	88	81
10	77	85
<b>average:</b>	<b>82</b>	<b>85</b>

Is model 2 better than model 1?

34

## Comparing systems: sample 3

split	model 1	model 2
1	84	87
2	83	86
3	78	82
4	80	86
5	82	84
6	79	87
7	83	84
8	83	86
9	85	83
10	83	85
<b>average:</b>	<b>82</b>	<b>85</b>

Is model 2 better than model 1?

35

## Comparing systems

split	model 1	model 2
1	84	87
2	83	86
3	78	82
4	80	86
5	82	84
6	79	87
7	83	84
8	83	86
9	85	83
10	83	85
<b>average:</b>	<b>82</b>	<b>85</b>

split	model 1	model 2
1	87	87
2	92	88
3	74	79
4	75	86
5	82	84
6	79	87
7	83	81
8	83	92
9	88	81
10	77	85
<b>average:</b>	<b>82</b>	<b>85</b>

What's the difference?

36

### Comparing systems

split	model 1	model 2	split	model 1	model 2
1	84	87	1	87	87
2	83	86	2	92	88
3	78	82	3	74	79
4	80	86	4	75	86
5	82	84	5	82	84
6	79	87	6	79	87
7	83	84	7	83	81
8	83	86	8	83	92
9	85	83	9	88	81
10	83	85	10	77	85
<b>average:</b>	<b>82</b>	<b>85</b>	<b>average:</b>	<b>82</b>	<b>85</b>
<b>std dev</b>	<b>2.3</b>	<b>1.7</b>	<b>std dev</b>	<b>5.9</b>	<b>3.9</b>

Even though the averages are same, the variance is different!

### Comparing systems: sample 4

split	model 1	model 2
1	80	82
2	84	87
3	89	90
4	78	82
5	90	91
6	81	83
7	80	80
8	88	89
9	76	77
10	86	88
<b>average:</b>	<b>83</b>	<b>85</b>
<b>std dev</b>	<b>4.9</b>	<b>4.7</b>

Is model 2 better than model 1?

37

38

### Comparing systems: sample 4

split	model 1	model 2	model 2 - model 1
1	80	82	2
2	84	87	3
3	89	90	1
4	78	82	4
5	90	91	1
6	81	83	2
7	80	80	0
8	88	89	1
9	76	77	1
10	86	88	2
<b>average:</b>	<b>83</b>	<b>85</b>	
<b>std dev</b>	<b>4.9</b>	<b>4.7</b>	

Is model 2 better than model 1?

39

### Comparing systems: sample 4

split	model 1	model 2	model 2 - model 1
1	80	82	2
2	84	87	3
3	89	90	1
4	78	82	4
5	90	91	1
6	81	83	2
7	80	80	0
8	88	89	1
9	76	77	1
10	86	88	2
<b>average:</b>	<b>83</b>	<b>85</b>	
<b>std dev</b>	<b>4.9</b>	<b>4.7</b>	

Model 2 is ALWAYS better

40

## Comparing systems: sample 4

split	model 1	model 2	model 2 - model 1
1	80	82	2
2	84	87	3
3	89	90	1
4	78	82	4
5	90	91	1
6	81	83	2
7	80	80	0
8	88	89	1
9	76	77	1
10	86	88	2
average:	83	85	
std dev	4.9	4.7	

How do we decide if model 2 is better than model 1?

41

## Statistical tests

### Setup:

- Assume some default hypothesis about the data that you'd like to *disprove*, called the **null hypothesis**
- e.g. model 1 and model 2 are not statistically different in performance

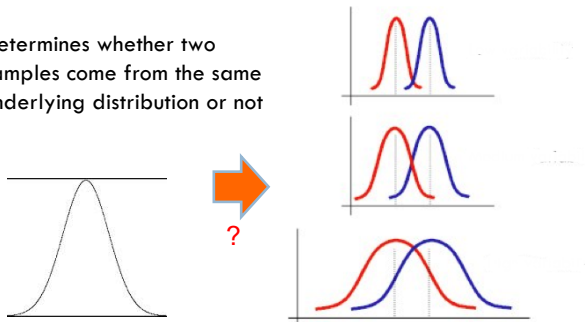
### Test:

- Calculate a test statistic from the data (often assuming something about the data)
- Based on this statistic, with *some probability* we can **reject the null hypothesis**, that is, show that it does not hold

42

## t-test

Determines whether two samples come from the same underlying distribution or not



43

## t-test

Null hypothesis: model 1 and model 2 accuracies are no different, i.e. come from **the same** distribution

Assumptions: there are a number that often aren't completely true, but we're often not too far off

Result: probability that the difference in accuracies is due to random chance (low values are better)

44

## Calculating t-test

For our setup, we'll do what's called a "paired t-test"

- The values can be thought of as pairs, where they were calculated under the same conditions
- In our case, the same train/test split
- Gives more power than the unpaired t-test (we have more information)

For almost all experiments, we'll do a "two-tailed" version of the t-test

Can calculate by hand or in code, but why reinvent the wheel: use excel or a statistical package

[http://en.wikipedia.org/wiki/Student's\\_t-test](http://en.wikipedia.org/wiki/Student's_t-test)

45

## p-value

The result of a statistical test is often a p-value

p-value: the probability that the null hypothesis holds. Specifically, if we re-ran this experiment multiple times (say on different data) what is the probability that we would reject the null hypothesis incorrectly (i.e. the probability we'd be wrong)

Common values to consider "significant": 0.05 (95% confident), 0.01 (99% confident) and 0.001 (99.9% confident)

46

## Comparing systems: sample 1

split	model 1	model 2
1	87	88
2	85	84
3	83	84
4	80	79
5	88	89
6	85	85
7	83	81
8	87	86
9	88	89
10	84	85
<b>average:</b>	<b>85</b>	<b>85</b>

Is model 2 better than model 1?

They are the same with:  $p = 1$

47

## Comparing systems: sample 2

split	model 1	model 2
1	87	87
2	92	88
3	74	79
4	75	86
5	82	84
6	79	87
7	83	81
8	83	92
9	88	81
10	77	85
<b>average:</b>	<b>82</b>	<b>85</b>

Is model 2 better than model 1?

They are the same with:  $p = 0.15$

48

### Comparing systems: sample 3

split	model 1	model 2
1	84	87
2	83	86
3	78	82
4	80	86
5	82	84
6	79	87
7	83	84
8	83	86
9	85	83
10	83	85
average:	82	85

Is model 2 better than model 1?

They are the same with:  $p = 0.007$

49

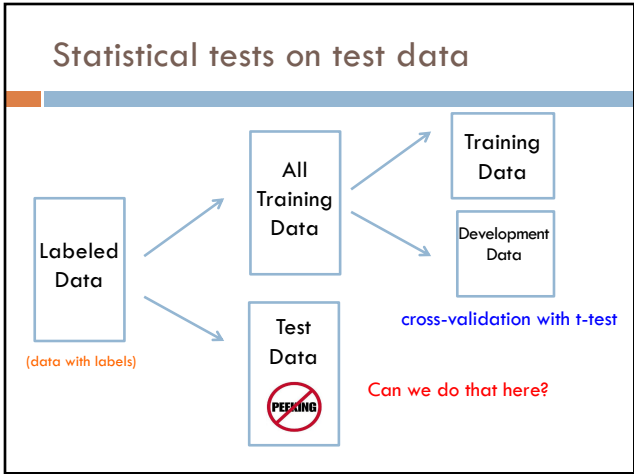
### Comparing systems: sample 4

split	model 1	model 2
1	80	82
2	84	87
3	89	90
4	78	82
5	90	91
6	81	83
7	80	80
8	88	89
9	76	77
10	86	88
average:	83	85

Is model 2 better than model 1?

They are the same with:  $p = 0.001$

50



51

### Bootstrap resampling

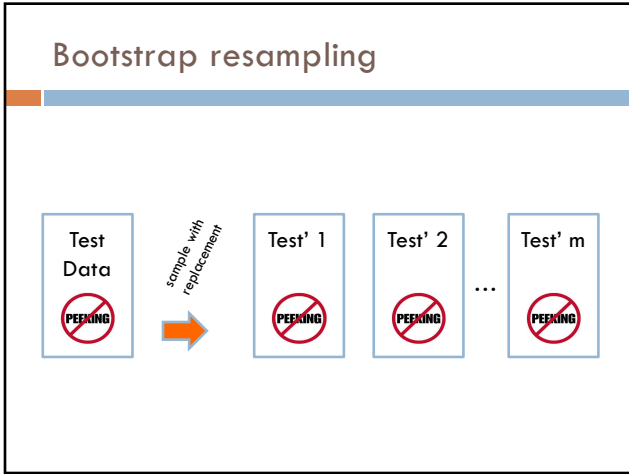
test set  $t$  with  $n$  samples

do  $m$  times:

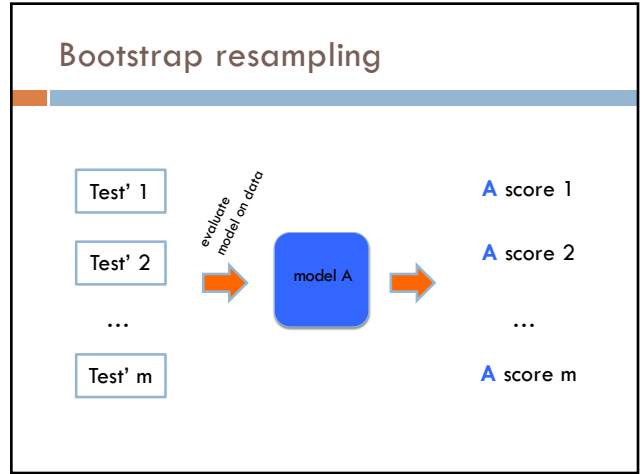
- sample  $n$  examples **with replacement** from the test set to create a new test set  $t'$
- evaluate model(s) on  $t'$

calculate t-test (or other statistical test) on the collection of  $m$  results

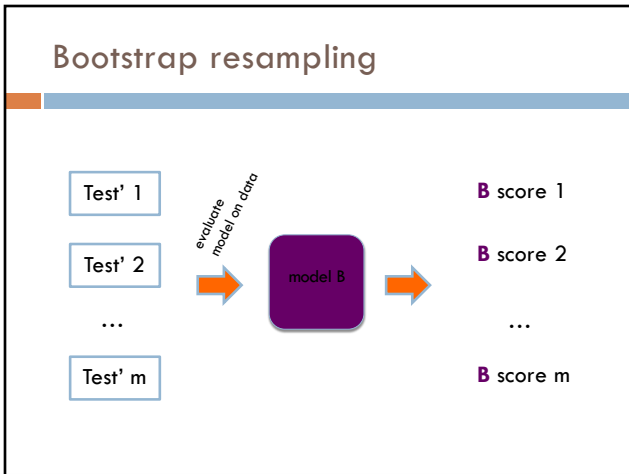
52



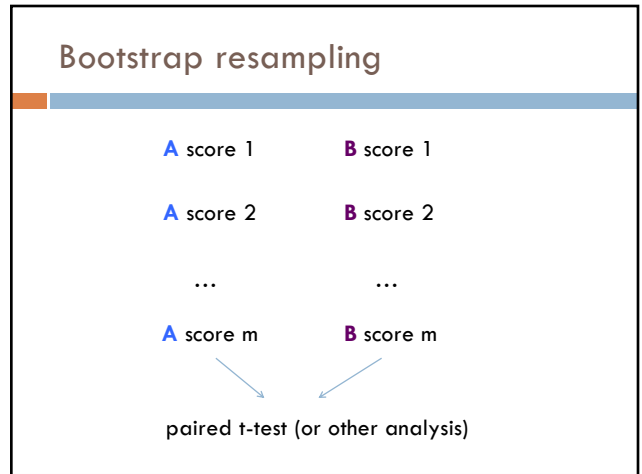
53



54



55



56

## Experimentation good practices

Never look at your test data!

During development

- ▣ Compare different models/hyperparameters on development data
- ▣ use cross-validation to get more consistent results
- ▣ If you want to be confident with results, use a t-test and look for  $p = 0.05$  (or even better)

For final evaluation, use bootstrap resampling combined with a t-test to compare final approaches