Evaluating Hypothesis and Experimental Design

Patricia J Riddle

Computer Science 760

Evaluating Hypothesis

Given **observed accuracy** of a hypothesis over a **limited sample of data**, how well does this **estimate it's accuracy** over additional examples?

Given that one hypothesis **outperforms another** over some **sample of data**, how probable is it that this hypothesis is more accurate in general?

When data is limited what is the best way to use this **data to both learn** a hypothesis and **estimate** its accuracy?

Estimating Hypothesis Accuracy

Estimating the accuracy with which it will classify future instances –

- also probable error of this accuracy estimate!!!
- (dice example)

A space of possible instances **X**.

Different instances in **X** may be encountered with different frequencies which is modeled by some unknown probability distribution **D**.

Notice **D** says nothing about whether **x** is a positive or negative instance – not looking at class yet

Learning Task

The learning task is to learn the target concept, **f**, by considering a space **H** of possible hypothesis.

Training examples of the target function **f** are provided to the learner by a trainer who draws each instance independently, according to the distribution **D** and who then forwards the instance **x** along with the correct target value **f(x)** to the learner.

Are instances ever really drawn independently?

Sample error

Sample error - the fraction of instances in some sample S that it misclassifies

$$error_s(h) \equiv \frac{1}{n} \sum_{x \in S} \delta(f(x), h(x))$$

Where

n is the number of samples in S, and

$\delta(f(x),h(x))$ is 1 if $f(x) \neq h(x)$ and 0 otherwise

True Error

True error - probability it will misclassify a single randomly drawn instance from the distribution D

$$error_D(h) \equiv \Pr_{x \in D}[f(x) \neq h(x)]$$

Where $\Pr_{x \in D}$ denotes that the probability is taken over the instance distribution D.

Sample error versus True error

Really want $error_D(h)$ but can only get $error_S(h)$.

How good an estimate of $error_D(h)$ is provided by $error_S(h)$?

Problems with Estimating Accuracy

Bias in Estimate

Variance in the Estimate

Bias in Estimate

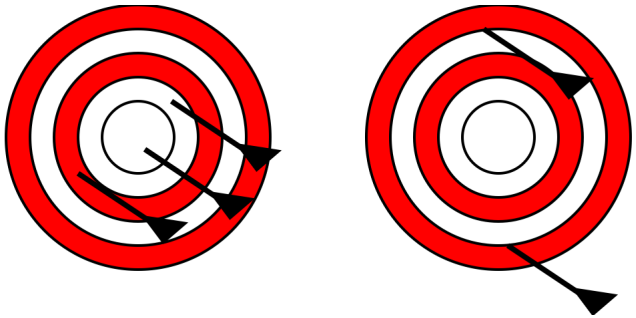
Observed accuracy of the learned hypothesis over the **training examples** is an **optimistically biased estimate** of hypothesis accuracy over future examples.

Especially likely when the learner considers a **very rich hypothesis space**, enabling it to **overfit** the training examples.

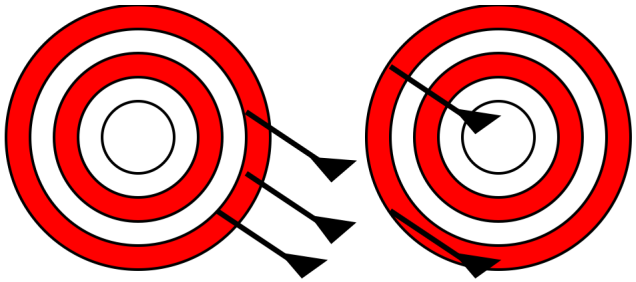
(what people are wearing example)

Typically we test the hypothesis on some set of **test examples chosen independently** of the training examples and the hypothesis.

Low Bias



High Bias



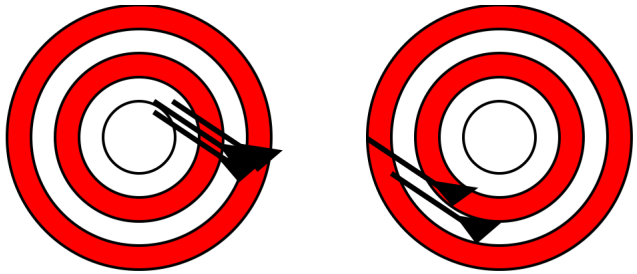
Variance in Estimate

Even if the hypothesis accuracy is measured over an unbiased set of test examples, the **measured accuracy** can still **vary** from **true accuracy**, depending on the makeup of the particular set of test examples.

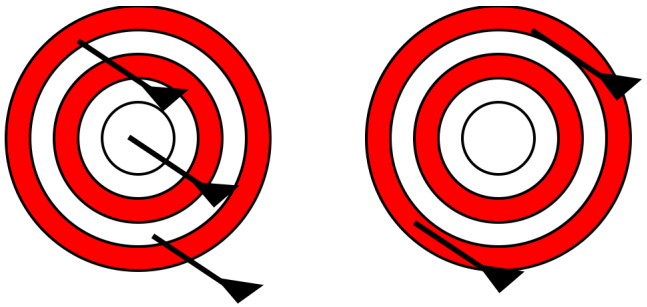
The **smaller** the **set** of test examples, the **greater** the expected **variance**.

(dice example again)

Low Variance



High Variance



Types of Bias

Machine Learning Bias

Systematic Error Bias

“Statistical” Bias

Machine Learning Bias

Every inductive learning algorithm must adopt a bias in order to generalize beyond the training data.

This is good and bad!

Systematic Error Bias

If there is systematic error in the training set, the learning algorithm cannot tell the difference between **systematic error** and **real signal** in the dataset.

can't tell the error from the signal

Therefore systematic error will also create a bias in the estimate.

Systematic error example - pull-down menus

Statistical Bias

Statistical Bias is the systematic error for a given sample size m.

"statistical bias" is the notion that

as the training set size gets smaller, then the error will go up.

Can we test for Bias?

Sort of

Statistical Bias Formula

Bias(A,m,x) = f´(x) - f(x), where

A is the learning algorithm,

m is the training set size,

x is a random example, and

f´ is the expected value of f, where the expectation is taken over all possible training sets of fixed size m.

$$f'(x) = \lim_{l \to \infty} \frac{1}{l} \sum_{i=1}^{l} f_{s_i}(x)$$

Variance

Variance(A,m,x) =E[$(f_S(x)-f'(x))^2$], where

f_S is a particular hypothesis learned on training set S.

Variance comes from

variation in the training data,

random noise in the training data, or

random behavior in the learning algorithm itself.

Not a Very Practical Method

What else can we do????

Error

So error is just made up of Bias and Variance.

$$Error(A,m,x)=Bias(A,m,x)^2+Variance(A,m,x)$$

Remember that the Bias includes “statistical bias”and Machine Learning Bias

it doesn’t include systematic error because it will be in the test set as well – it will look like signal

Also Bias is squared only because Variance is already squared

Sample Variance

Sample average

$$\bar{X} = (X_1 + \ldots + X_n) / n$$

Sample variance

$$S^2 = \frac{1}{n} \sum_{i=1}^{n} (X_i - \bar{X})^2$$

Calculated Bias

$$Error(A,m,x)=Bias(A,m,x)^2+Variance(A,m,x)$$

$$Bias(A,m,x)=sqrt(Error(A,m,x)-Variance(A,m,x))$$

This bias would include both statistical bias and ML bias (not systematical bias!!!)

Why can't we measure Systematic Bias????

Systematic error will only appear in the Error calculation if the test set does not have the systematic error (like if the sensor has been fixed)

Absolute and Relative ML Bias

Remember these definitions from 367 decision tree lectures

Absolute Bias - certain hypothesis are entirely eliminated from the hypothesis space - also called restriction of language bias

Relative Bias - certain hypothesis are preferred over others - also called preference or search bias

Effect of ML Bias on Stat Bias and Variance

ML Bias Absolute	Relative	Statistical Bias	Variance
appropriate	too strong	high	low
appropriate	ok	low	low
appropriate	too weak	low	high
inappropriate	too strong	high	low
inappropriate	ok	high	moderate
inappropriate	too weak	high	high

Four Important Sources of Error

Random variation in the selection of the test data

- (predicting stock market in 1929)
- Much more a problem as the test set becomes smaller - (Monday October 28th 1929)

Random variation in the selection of the training data

- Data only from the summer or only in a Bear market
- Same size factor

Randomness in the learning algorithm (e.g., initial weights)

- trying 2000 seeds and only one works well

Random classification error

- Human error (not machine error – must be random)

Dealing with Error

Good statistical test should not be fooled by these sources of variation.

To account for test-data variation and

random classification error,

the statistical procedure must consider the size of the test set and the consequences of changes in the test set.

To account for training-data variation and

internal randomness,

the statistical procedure must execute the learning algorithm multiple times and measure the variation in accuracy of the resulting classifiers.

What is Overfitting

Given a hypothesis space H, a hypothesis $h \in H$ is said to **overfit** the training data if there exists some alternative hypothesis $h' \in H$, such that h has a smaller error than h´ over the training examples, but h´ has a smaller error that h over the entire distribution of instances.

Not a very useful definition!

Other Phenomena

- Oversearching
- Feature Selection Problem

Oversearching

We use the term **oversearching** to describe the discovery by extensive search of a theory that is not necessarily over-complex but whose apparent accuracy is also misleading

Selecting models with lower performance (this means test set performance) as the size of the search space grows so must make sure do **not search too much**!!!!

Feature Selection Problem

A certain set of features seems best of the training set

But another set of features look better on the test set

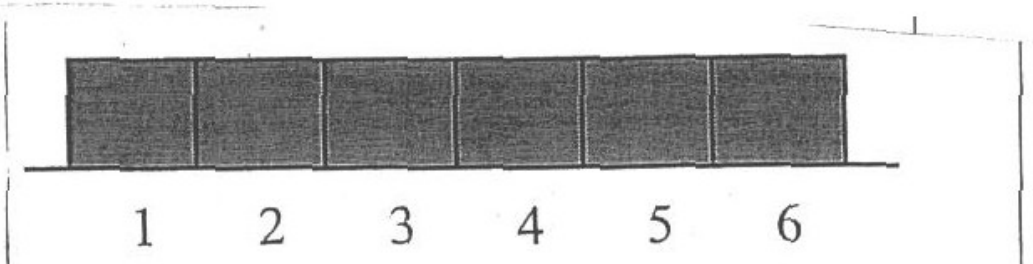
What causes Overfitting?

Many people think choosing too complex a hypothesis causes overfitting.

Why would complexity cause overfitting???

What about multiple comparisons?

Sampling Distributions for 1 die and 10 dice?

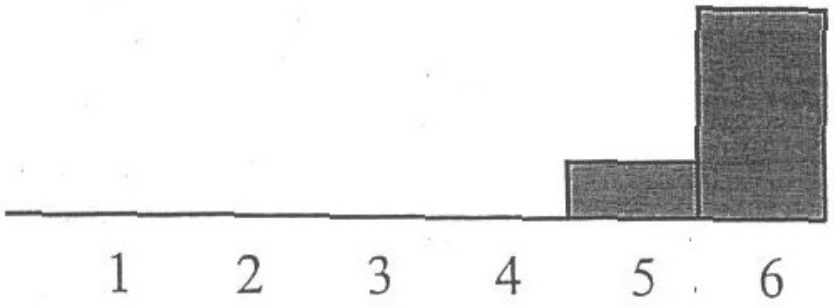


$E(X_i|H_0) = 3.5$

$p(X_i>5|H_0) = 0.167$

(a)

ampling distributions for one die and ten dice



Multiple Comparisons

- Cause overfitting, oversearching, feature selection problems
- Solutions
  - New test data
  - Bonferroni & Sidak (mathematical adjustment, assumes independence)
  - Cross validation - biased if k is to large because then the training sets are virtually the same - leave one out
  - Randomization tests - my favorite - drawback is time complexity - but to estimate p-values between .1 and .01 usually requires no more than 100-1000 trials

Multiple Comparisons Problem Increases

- Number of attributes goes up
- Number of Data Points does down

- Random Error Goes up == Signal getting more complex

10-fold Cross Validation

Break data into 10 sets of size n/10.

Train on 9 datasets and test on 1.

Repeat 10 times and take a mean accuracy and a standard deviation.

Also…

10-fold stratified cross validation has since been shown to have a lot of variance. SO need to do 10x10-fold cv to get good results. (Bouckaer) – different splits

Final classifier should be learned over the whole training set.

Randomisation Test

A **permutation test** (also called a randomization test, re-randomization test, or an exact test) is a type of statistical significance test in which a **reference distribution is obtained by calculating all possible values** of the test statistic under rearrangements of the labels on the observed data points.

Parametric versus Non-parametric

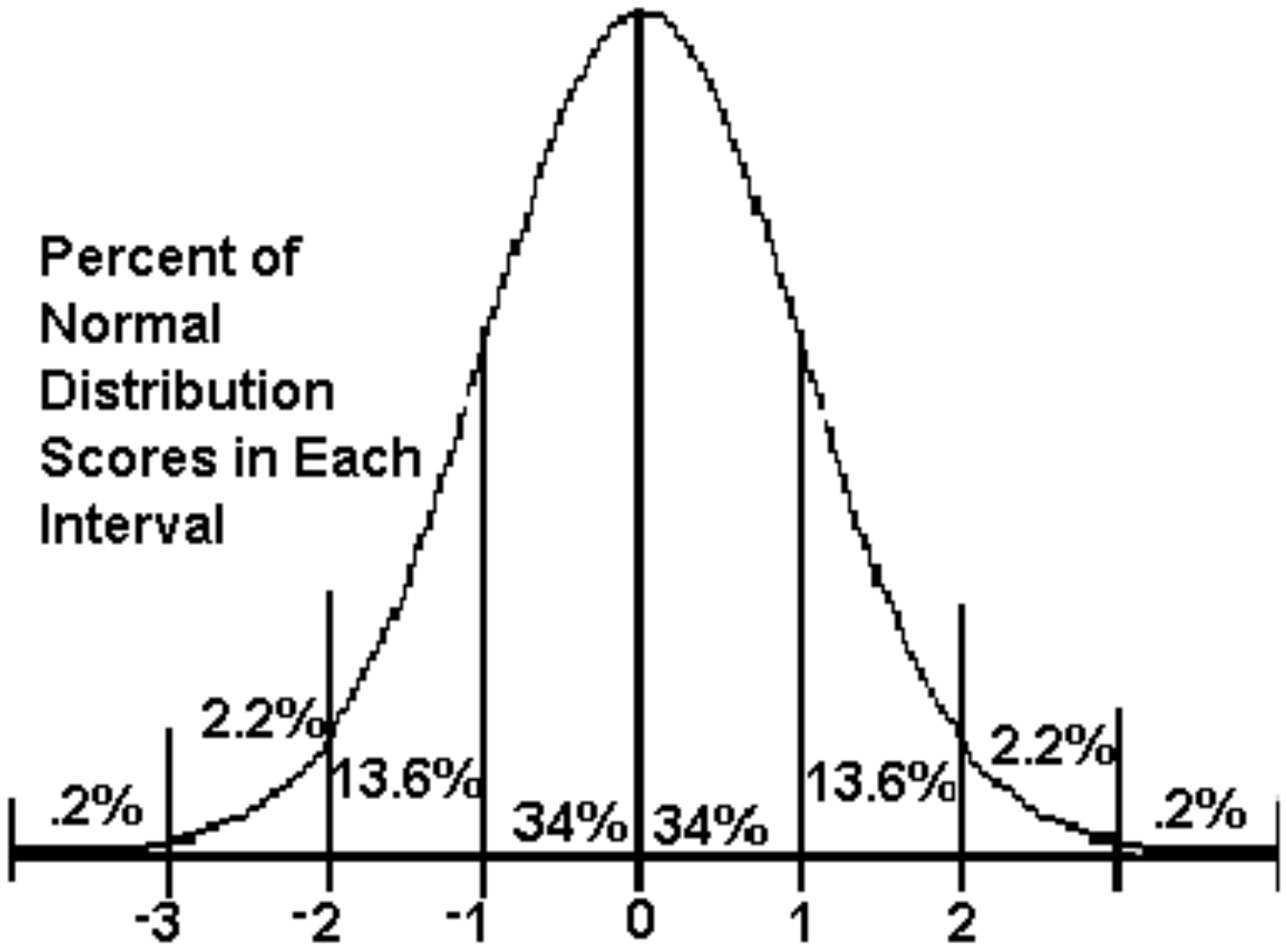
Parametric tests, such as those described in exact statistics, are exact tests when the **parametric assumptions are fully met**, but in practice the use of the term exact (significance) test is reserved for those tests that do not rest on parametric assumptions – non-parametric tests.

However, in practice most implementations of **non-parametric** test software use **asymptotical algorithms** for obtaining the significance value, which makes the implementation of the test non-exact.

ML Randomisation Test

1. Run your ML algorithm and get a value (error, accuracy, chi-sqaure)
2. Remove the target column from the training data (i.e., using awk)
3. Randomly shuffle the target column values using a random number generator
4. Reattach the column (the target should now be random)
5. Run your ML algorithm and get a value (error, accuracy, chi-sqaure)
6. Repeat line 2 and 5 until 100-1000 (or more) values are calculated.
7. Plot these numbers. (These numbers will be a normal distribution.)
8. Find out the confidence interval that your original value (line 1) gives you

This gives you a Normal Distribution



Now go back to your original dataset

- Now compare your results to the normal distribution you derived
  - The horizontal axis of your normal distribution is your value (like chi-sqaure)
  - The vertical axis is the number of occurrences of that value
  - This tells you the probability that your result is random for this dataset

Why I like Randomization tests

It tells you the probability of overfitting/ oversearching etc. for this particular dataset

No external assumptions are made

So if you have a particularly odd dataset it takes that into account

Rejected parts

Why accuracy and error are bad

- at least when learning rules

Confusion Matrix

	Then True	Then False	
IF True	a	b	N_{IT}
IF False	c	d	N_{IF}
	N_{TT}	N_{TF}	N

Let us look at Accuracy & Error

- Acc = a/N_{IT}
- Error = b/N_{IT}

Confidence, Support, Lift

- Support = P(if and then) = a/N
- Confidence = P(if and then)/P(if) = Support/P(if) = (a/N)(N_{IT}/N) = a/N_{IT}
- Lift = P(if and then) / P(if) P(then) = Support/P(if) P(then) = (a/N)/(N_{IT}/N)(N_{TT}/N) = (a/N)(N^2/N_{IT} * N_{TT})=aN/(N_{IT}*N_{TT})

Confusion Matrix Again

	Then CS-student	Then not CS-Student	
IF Male	20	30	N_{IT}
IF Female	2	1	N_{IF}
	N_{TT}	N_{TF}	N

Confusion Matrix Again Again

	Then CS-student	Then not CS-Student	
IF Male	20	30	N_{IT}
IF Female	2	1,000,000	N_{IF}
	N_{TT}	N_{TF}	N

Chi Square Formula

for 2x2 contigency table (confusion matrix)

$$X^2 = \frac{N(ad - bc)^2}{N_{IT} N_{IF} N_{TT} N_{TF}}$$

Why I like Chi-Squared or why I hate accuracy and error

- Chi-square covers the whole table
  - What if not(if) only happens 1% of the time
  - What if not(if) happens 99% of the time
    - Are these very different rules?

- If you have a rule that is 100% accurate it could be
  - 1/1, 10/10, 100000/100000
    - Are these all very different rules even though they give the same accuracy?

How is Multiple Comparison related to Complexity

- A complex hypothesis tends to cover a smaller subset to the dataset

- The smaller the subset the higher the chance of multiple comparisons

Why does Pruning Decision Trees Work?

- By pruning decision trees we are making the hypothesis space smaller (only small decision trees are allowed) so the effect of the multiple comparison's problem is reduced.

- Do I believe this?

Questions you should be able to answer

- What is the difference between Sample Error and True Error?
- What is Bias
- What is variance?
- What are the four sources of error?
- How do we minimise these sources?
- What is the real cause of overfitting?
- How does randomization testing help?

References

Machine Learning book, chapter 5

Machine Statistical Learning Bias Statistical Variance of Decision Tree, Thomas G Dietterich tgd@cs.orst.edu, Eun Bae Kong ebkong@cs.orst.edu

http://www.iiia.csic.es/~vtorra/tr-bias.pdf