

CS446 Introduction to Machine Learning (Fall 2015)
University of Illinois at Urbana-Champaign
<http://courses.engr.illinois.edu/cs446>

LECTURE 9: EVALUATION

Prof. Julia Hockenmaier
juliahmr@illinois.edu

Admin

Homework 1 is being graded.

Homework 2:

- Do not buy Matlab!!

- We have clarified Problem 1

- We have added a readme file for the Matlab part.

Project proposals:

- Submit on Compass by Thursday.

Recap: duals and kernels

Dual representation

Classifying \mathbf{x} in the primal: $f(\mathbf{x}) = \mathbf{w} \mathbf{x}$

\mathbf{w} = feature weights (to be learned)

$\mathbf{w} \mathbf{x}$ = dot product between \mathbf{w} and \mathbf{x}

Classifying \mathbf{x} in the dual: $f(\mathbf{x}) = \sum_n \alpha_n y_n \mathbf{x}_n \mathbf{x}$

α_n = weight of n -th training example (to be learned)

$\mathbf{x}_n \mathbf{x}$ = dot product between \mathbf{x}_n and \mathbf{x}

The dual representation is advantageous
when **#training examples** \ll **#features**
(requires fewer parameters to learn)

The kernel trick

- Define a feature function $\phi(\mathbf{x})$ which maps items \mathbf{x} into a higher-dimensional space.
- The kernel function $K(\mathbf{x}^i, \mathbf{x}^j)$ computes the inner product between the $\phi(\mathbf{x}^i)$ and $\phi(\mathbf{x}^j)$

$$K(\mathbf{x}^i, \mathbf{x}^j) = \phi(\mathbf{x}^i)\phi(\mathbf{x}^j)$$

- Dual representation: We don't need to learn \mathbf{w} in this higher-dimensional space. It is sufficient to evaluate $K(\mathbf{x}^i, \mathbf{x}^j)$

Constructing kernels

We have looked at a few examples of basic kernel functions (e.g. quadratic/polynomial kernels)

We have looked at ways to construct more complex kernel functions.

Kernels over (finite) sets

X, Z : subsets of a finite set D with $|D|$ elements

$k(X, Z) = |X \cap Z|$ (the number of elements in X and Z)
is a valid kernel:

$k(X, Z) = \phi(X)\phi(Z)$ where $\phi(X)$ maps X to a bit vector of length $|D|$
(i -th bit: does X contains the i -th element of D ?).

$k(X, Z) = 2^{|X \cap Z|}$ (the number of subsets shared by X and Z)
is a valid kernel:

$\phi(X)$ maps X to a bit vector of length $2^{|D|}$
(i -th bit: does X contains the i -th subset of D ?)

Statistical hypothesis testing

Why hypothesis testing?

Q: If $\text{Accuracy}(A) > \text{Accuracy}(B)$, can we conclude that classifier A is better than B?

A: No, not necessarily. Only if the difference between $\text{Accuracy}(A)$ and $\text{Accuracy}(B)$ is unlikely to arise by chance.

Hypothesis testing

We have a **hypothesis** H that we wish to show is true.
(H = “There is a difference between A and B”)

We have a **statistic** M that measures the difference between A and B, and we have measured **a value** m of M in our data.
But m itself doesn’t tell us whether H is true or false.

Instead, we estimate how likely m were to arise if the opposite of H (= the ‘**null hypothesis**’, H_0) was true.
(H_0 = “There is no difference between A and B”).

If $P(M \geq m | H_0) < p$, we can **reject** H_0 with p-value p

Rejecting H_0

- H_0 defines a distribution $P(M | H_0)$ over some statistic M
(e.g. M = the difference in accuracy between A and B)
- **Select a significance value S** (e.g. 0.05, 0.01, etc.)
You can only reject H_0 if $P(M=m | H_0) \leq S$
- Compute the **test statistic m** from your data
e.g. the average difference in accuracy over N folds
- Compute $P(M \geq m | H_0)$
- **Reject H_0 with p -value $p \leq S$ if $P(M \geq m | H_0) \leq S$**
Caveat: the p -value corresponds to $P(m | H_0)$, *not* $P(H_0 | m)$

p -Values

Commonly used p -values are:

- **0.05**: There is a 5% ($1/20$) chance to get the observed results under the null hypothesis.

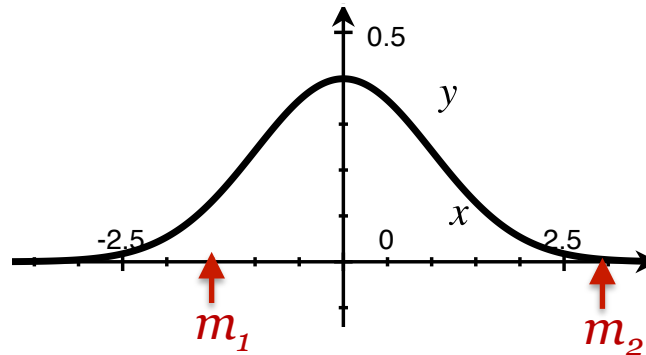
Corollary: If you run 20 or more experiments, at least one of them will yield results that fall in the “statistically significant range” with $p=0.05$, even if the null hypothesis is actually true.

- **0.01**: There is a 1% ($1/100$) chance to get the observed results under the null hypothesis.

Null hypothesis

Null hypothesis:

We assume the data comes from a (normal) distribution $P(M | H_o)$ with mean $\mu=0$ and (unknown) variance σ^2/N .



From the data (sample) $X = \{x^1 \dots x^N\}$, we compute the **sample mean** $m = \sum_i x^i / N$

How likely is it that m came from $P(M | H_o)$?

For m_1 : very likely. For m_2 : pretty unlikely

Confidence intervals

One-tailed test:

Test whether the accuracy of A is higher than B with probability p

Two-tailed test:

Test whether the accuracies of A and B are different (lower or higher) with probability p

This is the stricter test.

Confidence intervals

One-tailed test:

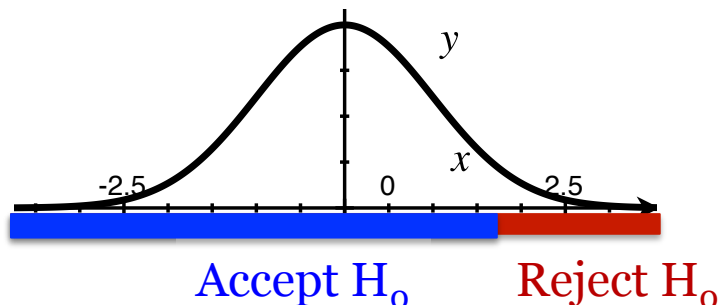
We fail to reject H_0 if m is inside the asymmetric $100(1-p)$ percent confidence interval $(-\infty, a)$

Two-tailed test:

We fail to reject H_0 if m lies in the symmetric $100(1-p)$ percent confidence interval $(-a, +a)$ around the mean.

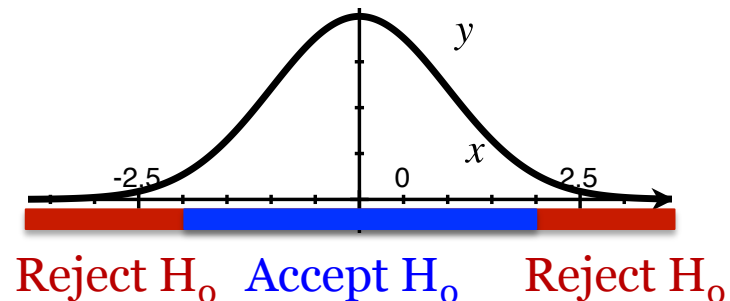
$p=0.05\%$; Confidence 95%

One-tailed test



$p=0.05\%$; Confidence 95%

Two-tailed test



Hypothesis tests to evaluate classifiers

Paired t-test:

Compare the performance of two classifiers on N test sets (e.g. N -fold cross-validation).

Uses the t-statistic to compute confidence intervals.

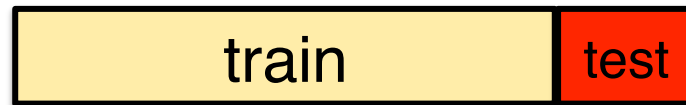
McNemar's test:

Compare the performance of two classifiers on N items from a single test set.

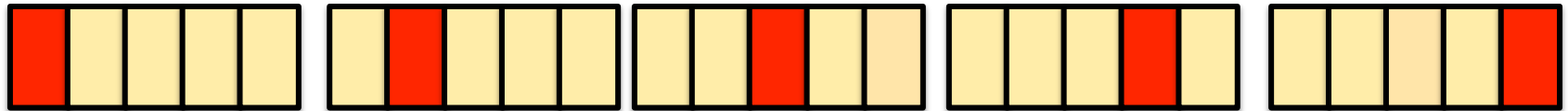
N-fold cross validation: Paired t-test

N-fold cross validation

Instead of a single test-training split:



- Split data into N equal-sized parts



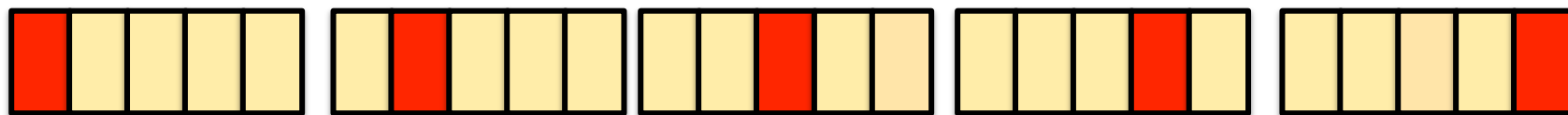
- Train and test N different instances of the same classifier
- This gives N different accuracies

Evaluating N-fold cross validation

	test set 1	test set 2	test set 3	test set 4	test set 5
A	80%	82%	85%	78%	85%
B	81%	81%	86%	80%	88%
$\text{diff}(A-B)$	-1%	+1%	-1%	-2%	-3%

The **paired t-test** tells us whether there is a (statistically significant) difference between the accuracies of classifiers A and B, based on their difference in accuracy on N different test sets.

Paired t-test for cross-validation



Two different classifiers, A and B are trained and tested using N-fold cross-validation

For the n -th fold:

$accuracy(A, n), accuracy(B, n)$

$diff_n = accuracy(A, n) - accuracy(B, n)$

Null hypothesis: $diff$ comes from a distribution with mean (expected value) = 0.

Paired t-test

Null hypothesis (H_0 ; to be rejected), informally:

There is no difference between A and B's accuracy.

- Statistically, we treat $\text{accuracy}(A)$ and $\text{accuracy}(B)$ as random variables drawn from some distribution.
- H_0 says that $\text{accuracy}(A)$ and $\text{accuracy}(B)$ are drawn from the same distribution.
- If H_0 is true, then the expected difference (over all possible data sets) between their accuracies is 0.

Null hypothesis (H_0 ; to be rejected), formally:

The difference between $\text{accuracy}(A)$ and $\text{accuracy}(B)$ on the same test set is a random variable with mean = 0.

$$H_0: E[\text{accuracy}(A) - \text{accuracy}(B)] = E[\text{diff}_D] = 0$$

Paired t-test

Null hypothesis (H_0 ; to be rejected), formally:
The difference between accuracy(A) and accuracy(B) on the same test set is a random variable with mean = 0.

$$H_0: E[\text{accuracy}(A) - \text{accuracy}(B)] = E[\text{diff}_D] = 0$$

- $E[\text{diff}_D]$ is the expected value (mean) over all possible data sets. We don't (can't) know that quantity.
- But N -fold cross-validation gives us N samples of diff_D

We can ask instead: **How likely are these N samples to come from a distribution with mean = 0?**

Paired t-test

Paired t-test: The accuracy of A on test set i is paired with the accuracy of B on test set i

Assumption: Accuracies are drawn from a normal distribution (with unknown variance)

Null hypothesis: The accuracies of A and B are drawn from the same distribution.

Hence, the *difference* of the accuracies on test set i comes from a normal distribution with mean = 0

Alternative hypothesis: The accuracies are drawn from two different distributions: $E[\text{diff}] \neq 0$

Paired t-test

Given: a sample of **N observations**

We assume these come from a normal distribution with fixed (but unknown) mean and variance

- Compute the **sample mean** and **sample variance** for these observations
- This allows you to compute the **t-statistic**.
- The **t-distribution for $N-1$ degrees of freedom** can be used to estimate how likely it is that the true mean is in a given range

Reject H_0 at significance level p if the t-statistic does not lie in the interval $(-t_{p/2, n-1}, +t_{p/2, n-1})$.

There are tables where you can look this up

Computing the t-statistic

Difference in accuracy on the n -th test set:

$$\text{diff}_n = \text{Accuracy}_n(\text{A}) - \text{Accuracy}_n(\text{B})$$

Sample mean m of diff_D , based on N samples of diff_D :

$$m = \frac{1}{N} \sum_{n=1}^N \text{diff}_n$$

Sample standard deviation S of diff_D :

$$S = \sqrt{\frac{\sum_{n=1}^N (\text{diff}_n - m)^2}{N-1}}$$

t-statistic for N samples of diff_D :

$$t = \frac{\sqrt{N} \cdot m}{S}$$

Can we reject H_0 ?

1. Compute the t-statistic t for your N samples.
2. Define a p-value $p \in \{0.05, 0.01, 0.001\}$.
3. Look up $t_{p/2, N-1}$ for $N-1$ degrees of freedom (df)
4. If $t > t_{N-1, p}$: Reject H_0 with p-value p

For our example:

	test set 1	test set 2	test set 3	test set 4	test set 5
A	80%	82%	85%	78%	85%
B	81%	81%	86%	80%	88%
$\text{diff}(A-B)$	-1%	+1%	-1%	-2%	-3%

$$m = (-1 + 1 - 1 - 2 - 3)/5 = -6/5 = -1.2$$

$$S = \sqrt{\frac{(-2.2)^2 + 2.2^2 + (-2.2)^2 + (-3.2)^2 + (-4.2)^2}{4}} \approx 3.256$$

Our t-statistic $t = -0.824$

With $p=0.05$ and $N-1 = 4$: $t_{0.025,4} = 2.776$

We cannot reject H_0 : t is between $-t_{0.025,4}$ and $+t_{0.025,4}$

$$-t_{0.025,4} = -2.776 < t = -0.824 < +t_{0.025,4} = 2.776$$

Summary t-test

The t-test can be used to to compare two classifiers on N-fold cross-validation.

Caveat: N should be at least 30.

Alternative: 5x2 Cross-validation

A single test set: McNemar's test

McNemar's test

Compares classifiers A and B on **a single test set**.

Considers the number of test items where either A or B make errors:

n_{11} : number of items classified correctly by both A and B

n_{00} : number of items misclassified by both A and B

n_{01} : number of items misclassified by A but not by B

n_{10} : number of items misclassified by B but not by A

Null hypothesis:

A and B have the same error rate. Hence, **$n_{01} = n_{10}$**

McNemar's test

Observed data:

n_{00}	n_{01}
n_{10}	n_{11}

Expected counts under H_0 :

n_{00}	$(n_{01} + n_{10})/2$
$(n_{01} + n_{10})/2$	n_{11}

Compute the χ^2 statistic

$$\chi^2 = \frac{(|n_{01} - n_{10}| - 1)^2}{n_{01} + n_{10}}$$

McNemar's test

Two-tailed test:

- Reject H_0 with $p=0.05$ if $\chi^2 > \chi_{.05}^2 = 3.84$
- Reject H_0 with $p=0.01$ if $\chi^2 > \chi_{.05}^2 = 6.63$

One-tailed test:

- Reject H_0 with $p=0.05$ if $\chi^2 > \chi_{.05}^2 = 2.71$
- Reject H_0 with $p=0.01$ if $\chi^2 > \chi_{.05}^2 = 5.43$

McNemar's test

McNemar's test is used to compare the performance of two classifiers on the same test set.

This test works if there are a large number of items on which A and B make different predictions.

Today's key concepts

Using significance tests to compare the performance of two classifiers:

t-test (Cross-validation)

McNemar's test (single test set)