Machine Learning – Scribe 7

March 2/4, 2004
Yinan Zhang, Hongyin Cui

Content

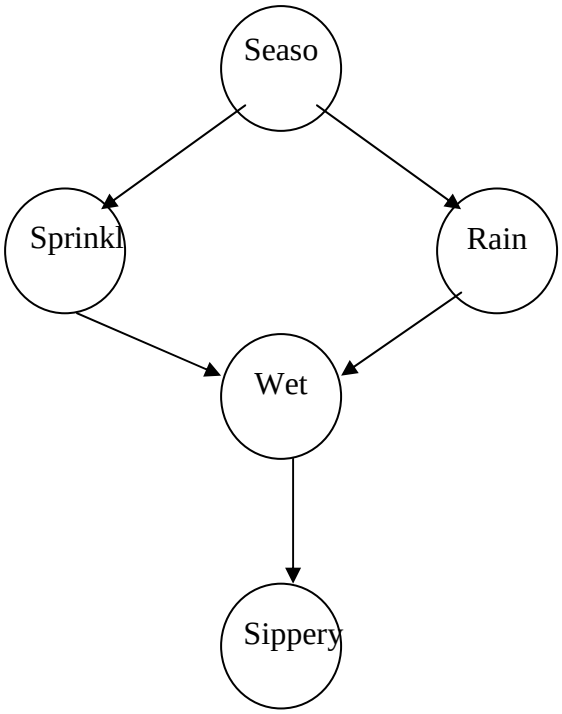
- Bayes Net
  - Inferring Causal Structure (Continued) - Rules for Orientation
- Network Traffic Modeling (presented by Leo Chen)
  - Bayesian Net (causal graph): B-Course, Tetrad
  - Prediction: ARMA, SARIMA
  - Data clustering: AutoClass, k-means
- Evaluating Hypotheses
  - Motivation
  - Definitions of sample error and true error
  - Estimating the true error
  - Confidence Intervals for Discrete-Valued Hypotheses
  - Some basic definitions and facts from statistics

Bayes Net

Inferring Causal Structure (Continued) - Rules for Orientation

1. Given a-> b, b-c, add b->c if a, c are not linked (no new collider, aka. no new V-structure).
2. Given a->c->b, a-b, add a->b (no cycle).
3. Given a-c->d, c->d->b, and a-b, add a->b if c, d aren't linked (no cycle & no new V-structure).
4. Given a-c->b, a-d->b, and a-b, add a->b if c, d aren't linked (no cycle & no new V-structure).

The sprinkler example revisited:



Known: both Sprinkler and Rain are the parents of Wet.

Given Sprinkler->Wet (or Rain->Wet), and Wet--Slippery, add Wet->Slippery to avoid a new V-Structure.

Network Traffic Modeling (presented by Leo Chen)

To accomplish more efficient performance and better understanding for a Vancouver-based network, Leo Chen is using some techniques and software to model and predict the network traffic. Figure-1 shows the network in a map. Figure-2 shows some raw data extracted from the database. Each circle in Figure-1 represents a location or system with a unique *Sys_id*. Each conversation group in the network has a same *Caller* number. From the raw data in Figure-2, we see that a group may involve multiple systems (*Sys_id*). Each column in the raw data represents a feature. The traffic pattern and some insight behind the raw data are desired to the network company.

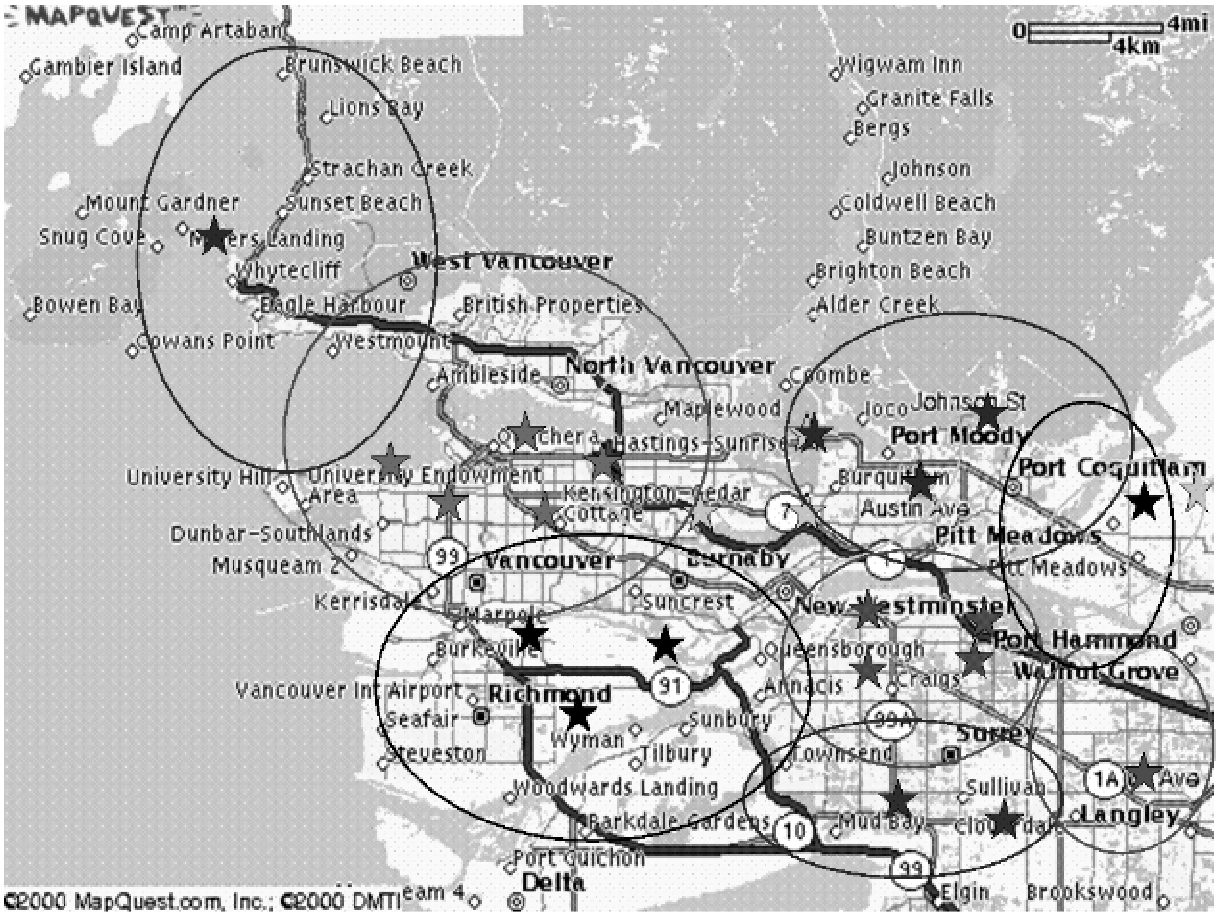


Figure-1.

Date	Time	Ms	Duration	Sys_id	Chl_id	Caller	Callee	C_type	C_state	Multi
2003-03-20	00:00:01	450	3730	8	4	6155	1801	0	0	0
2003-03-20	00:00:01	469	3730	6	7	6155	1801	0	0	0
2003-03-20	00:00:01	560	3730	3	7	6155	1801	0	0	0
2003-03-20	00:00:01	570	3730	2	7	6155	1801	0	0	0
2003-03-20	00:00:01	640	3730	1	7	6155	1801	0	0	0
2003-03-20	00:00:01	880	5260	9	6	13314	251	0	0	0
2003-03-20	00:00:01	910	5260	7	6	13314	251	0	0	0
2003-03-20	00:00:01	970	5260	6	8	13314	251	0	0	0
2003-03-20	00:00:01	980	2520	7	7	13911	418	0	0	0
2003-03-20	00:00:02	29	5270	4	2	13314	251	0	0	0
2003-03-20	00:00:02	109	5260	2	8	13314	251	0	0	0
2003-03-20	00:00:02	139	5270	1	8	13314	251	0	0	0
2003-03-20	00:00:02	9	2510	6	1	13911	418	0	0	0
2003-03-20	00:00:02	149	2510	2	9	13911	418	0	0	0
2003-03-20	00:00:05	289	3560	8	5	6011	2035	0	0	0
2003-03-20	00:00:05	309	3550	6	3	6011	2035	0	0	0
2003-03-20	00:00:05	389	3560	3	2	6011	2035	0	0	0
2003-03-20	00:00:05	449	3550	2	2	6011	2035	0	0	0
2003-03-20	00:00:05	480	3550	1	9	6011	2035	0	0	0
2003-03-20	00:00:05	550	3440	1	12	7614	945	0	0	0
2003-03-20	00:00:05	550	3440	2	3	7614	945	0	0	0
2003-03-20	00:00:05	949	9780	6	4	15840	418	0	0	0
2003-03-20	00:00:05	959	9780	7	2	15840	418	0	0	0
2003-03-20	00:00:06	679	3040	2	6	13931	471	0	0	0
2003-03-20	00:00:06	709	3040	1	2	13931	471	0	0	0
2003-03-20	00:00:06	130	9780	2	4	15840	418	0	0	0
2003-03-20	00:00:08	109	6640	9	2	13420	251	0	0	0
2003-03-20	00:00:08	179	6630	7	3	13420	251	0	0	0
2003-03-20	00:00:08	200	6640	6	5	13420	251	0	0	0
2003-03-20	00:00:08	270	6630	4	5	13420	251	0	0	0
2003-03-20	00:00:08	329	6640	1	4	13420	251	0	0	0
2003-03-20	00:00:08	340	6640	2	7	13420	251	0	0	0

Figure-2

Bayesian Network (causal gragh)

Since Bayesian Network is capable of inferring the causalities from the data, two existing software (B-Course and Tetrad) are used to construct a Bayesian Network to model the user's behavior. Based on the raw data, user behavior data are generated. The following shows some behavior data examples.

Hourly TalkGroup Behavior Characteristics

TalkGroup	Agency	Date	Hour	NC	NMC	AD	MD	AS	MS	AR	MR
0	1	20030301	0	26	0	413846	5920	1	1	413846	5920
0	1	20030301	1	25	0	40196	5660	1	1	40196	5660
0	1	20030301	2	24	0	415917	5800	1	1	415917	5800
0	1	20030301	3	24	0	398333	4610	1	1	398333	4610
0	1	20030301	4	24	0	399708	5270	1	1	399708	5270

TalkGroup Peak Hour Behaviors

TalkGroup	Agency	Date	Hour	NC	NMC	AD	MD	AS	MS	AR	MR
0	1	20030301	0	26	0	413846	5920	1	1	413846	5920
9	5	20030301	9	1	1	4860	4860	11	11	53460	53460
38	1	20030301	0	9	0	3481.11	7900	1	1	3481.11	7900
113	1	20030301	0	3	0	190667	2440	1	1	190667	2440
169	5	20030301	3	86	86	546605	18150	2	2	1087209	36300

B-Course - This is a web-based data analysis tool for Bayesian modelling. It is able to do dependency and classification modelling. You can find it in http://b-course.hiit.fi. Figure-3 and Figure-4 show the results of dependency modelling based on the user behaviour data.

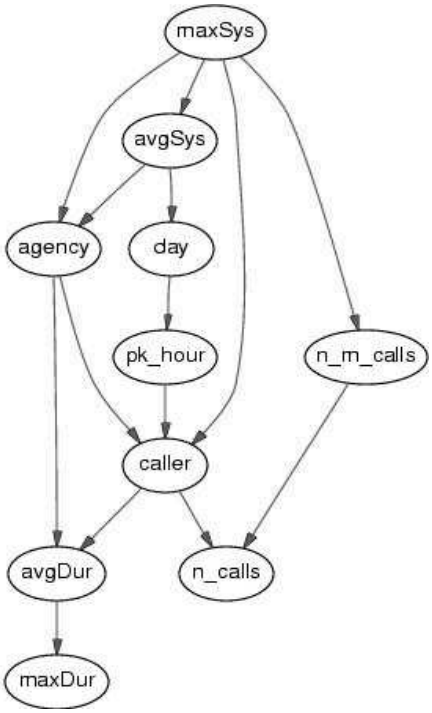


Figure-3. User Peak Hour Behaviour

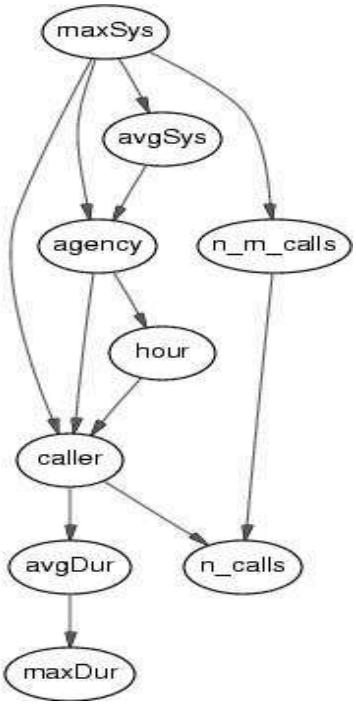


Figure-4. User Behaviour

From the above causal graph, it is obvious that *n_calls* is irrelevant with *avgDur* given *caller*.

Tetrad – This is a tool for learning causal structure from statistical data. You can find it in http://www.phil.cmu.edu/projects/tetrad. The causal graph generated by Tetrad may have some edges with bi-direction arrow; it is because it can not determine the direction.

Prediction

To predict user's future behavior, ARMA and SARIMA models are used. ARMA is based on the user's behavior in the previous 1 or 2 days. SARIMA is based on the user's seasonal behavior in the previous 1 or 2 weeks. Figure-5 shows the result of SARIMA.

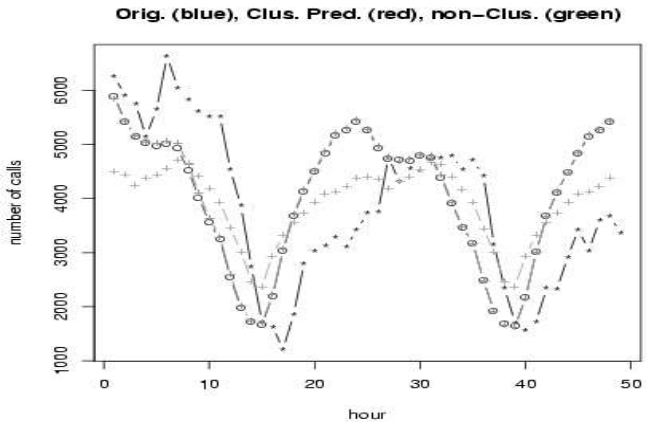


Figure-5.

Data Clustering

If users can be categorized into clusters and behavior pattern of each cluster can be discovered, then it will make it easier to model or calculate the network traffic when adding more new users. This motivates to cluster users into groups with two tools, AutoClass and K-means.

AutoClass – It clusters users into different groups given the user behavior data. Figure-6 shows the behavior pattern of each group after clustering.

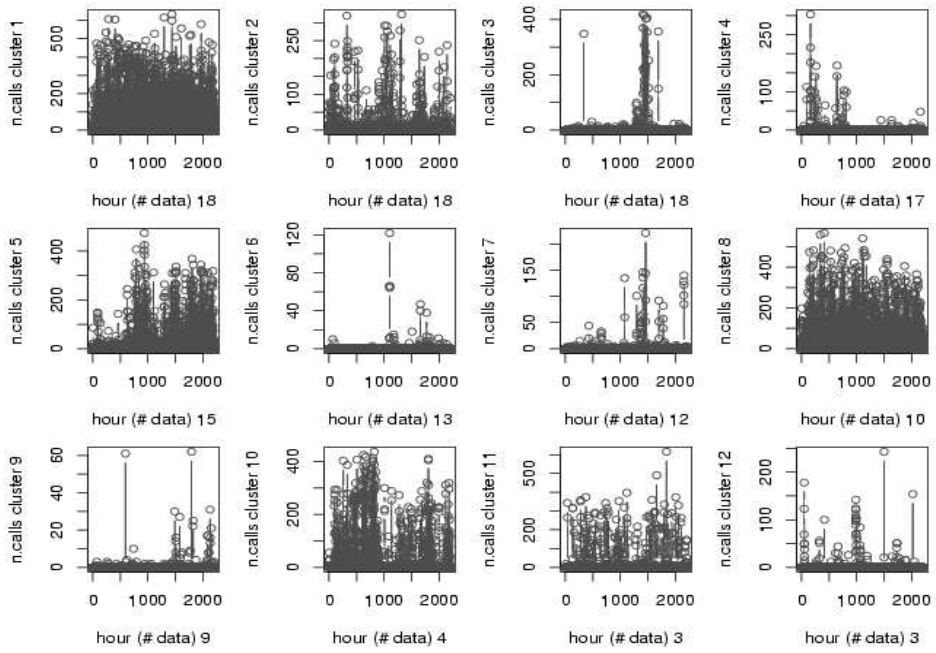


Figure-6.

K-means – This is a popular algorithm for clustering. The basic idea is:

- Form initial *k* non-empty random clusters
- Re-assign clusters based on distance (Euclidean distance) from centroids
- Update the centroids
- Repeat until converge

Figure-7 shows the behaviour pattern of each group after clustering.

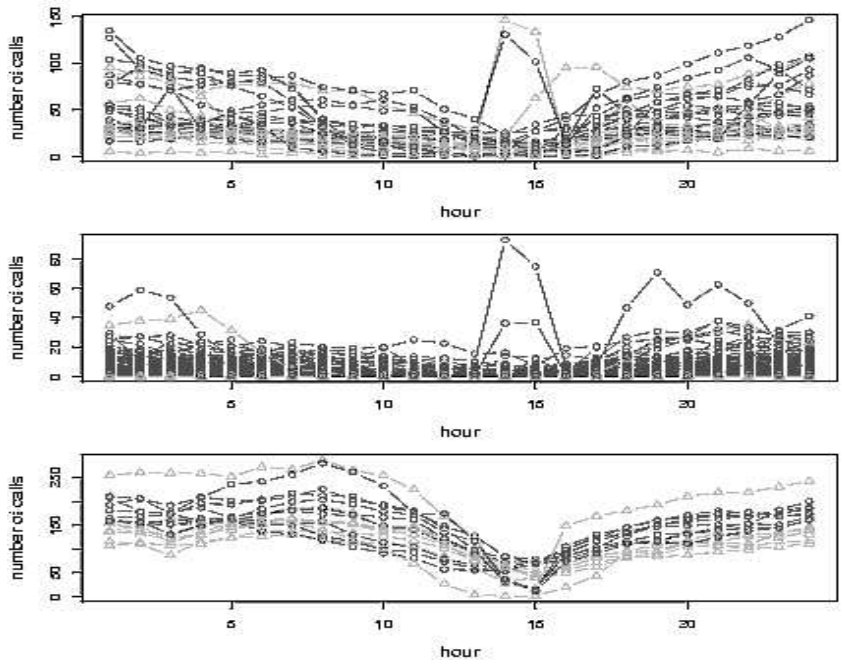


Figure-7

Evaluating Hypotheses

Motivation

1. It's important to understand the accuracy of the learned hypotheses if learning is based on a limited-size database.
2. Evaluating hypotheses is an integral component of many learning methods. For example, in post-pruning decision trees to avoid overfitting, it's critical to understand the likely errors of estimating the accuracy of both the pruned and unpruned tree.

Two questions we are interested:
1. Given a hypothesis h and a data sample containing n examples drawn at random according to the distribution D, what is the best estimate of the accuracy of h over future instances drawn from the same distribution.
2. What is the probable error in this accuracy estimate?

Definitions of sample error and true error

The true error (errorD(h)) of hypothesis h with respect to target function f and distribution D, is the probability that h will misclassify an instance drawn at random according to D.

$$error_{\mathcal{D}}(h) \equiv \Pr_{x \in \mathcal{D}}[f(x) \neq h(x)] \tag{1.1}$$

where $\Pr_{x \in \mathcal{D}}$ denotes the probability taken over the instance distribution D.

The sample error (errorS(h)) of hypothesis h with respect to target function f and data sample S is the proportion of example h misclassifies

$$error_S(h) \equiv \frac{1}{n} \sum_{x \in S} \delta(f(x) \neq h(x)) \tag{1.2}$$

Where n is the number of examples in S, and the quantity $\delta(f(x), h(x))$ is 1 if $f(x) \neq h(x)$, and 0 otherwise.

errorD(h) is what we wish to know, and errorS(h) is what we can measure. It's important to determine how accurate errorS(h) is as an estimator of errorD(h).

Example:
If an hypothesis h misclassifies 12 of the 40 examples in S, errorS(h) = 12/40 = 0.3

Two key difficulties of estimating the accuracy of a learned hypothesis given a limited set of data:

1. Bias in the estimate: A learned hypothesis h derived from the given training samples S is

often a poor classifier over future examples ("selective perception") especially when a rich hypothesis space is taken into account.

$$bias \equiv E[error_S(h)] - error_D(h) \quad (1.3)$$

Thus, h and S must chosen independently. Usually, a set of test examples chosen independently of S is needed.

2. Variance in the estimate: errorS(h) can still vary from errorD(h) even if the accuracy, or the error, of h is measured over an unbiased set of test examples independent of S. The smaller the set of test examples, the greater the expected variance.

Estimating the true error

Back to the equation 1.3, the estimation bias of an estimator Y for an arbitrary parameter p is

E[Y]-p

If the estimation bias is zero, we say that Y is unbiased estimator for p. In another words, it means that the average of many random values of Y generated by repeated random experiments converges toward p.

Suppose five experiments are performed, and each experiment is based on n samples drawn randomly and independently according to the probability distribution. The sample errors of them are -20%, 0%, 20%, 10%, and 40%, respectively. If we use the mean which is 10% as our estimator, and the true error is also 10%, then we say our estimator is unbiased.

Also suppose two different sets of sample errors e1:{10%, 10%, 10%} and e2:{20%, 10%, 0%}. Despite the mean values of both e1 and e2 are the same, e1 is preferred because it has a smaller variance than e2. This introduces what is called "bias-variance trade-off."

- If we prefer an unbiased estimator, then its variance may be large.
- If we prefer no variance, then the estimator may be biased.

Confidence Intervals for Discrete-Valued Hypotheses

Suppose in an experiment, we wish to estimate the true error for some discrete-valued hypothesis h, based on its observed sample error over a sample S, where

- S contains n examples drawn independent of h and each other according to the probability distribution D.
- $n \geq 30$
- hypothesis h commits r errors over these n examples, ie. errorS(h) = r/n.
- $n*errors(h)*(1-errorS(h)) \geq 5$.

The statistical theory enables us to make the following assertions:

1. Given no other information, the most probable value of errorD(h) is errorS(h).
2. With approximately 95% probability, the true error errorD(h) lies in the interval

$$error_S(h) \pm 1.96\sqrt{\frac{error_S(h)(1-error_S(h))}{n}} \quad (1.4)$$

Example:
If a set of sample S, which contains 40 examples, has a sample error of 10%, then we expect errorD(h) lies in the interval 0.1±(1.96*0.0047) = 0.100±0.093.If we do another experiment with an independent set S' of 40 examples, we expect errorS'(h) to be different from errorS(h) due to the random differences in the makeup of S and S'. However, in general, if we repeated this experiment over and over, we would find for approximately 95% of these experiments, the calculated interval would contain the true error.

The constant 1.96 is used if we desire a 95% confidence interval. In general, a different constant, Z_N, is used to calculate the N% confidence interval. The general expression for approximate N% confidence intervals for errorD(h) is

$$error_S(h) \pm z_N\sqrt{\frac{error_S(h)(1-error_S(h))}{n}} \quad (1.5)$$

where

$N\%$:	50%	68%	80%	90%	95%	98%	99%
z_N:	0.67	1.00	1.28	1.64	1.96	2.33	2.58

Notice that Z_N increases while N% increases, and it means that the more confident we desire to be, the larger the confidence interval we get because we reduce the probability with which we demand that errorD(h) fall into the interval.

Some basic definitions and facts from statistics

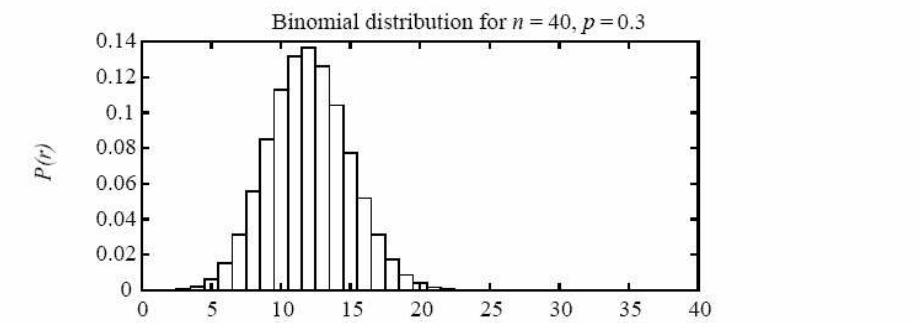
This part introduces some basic notions from statistics and sampling theory. A basic familiarity with these concepts is important to understanding how to evaluate hypotheses and learning algorithm. Even more important, these notions provide an important conceptual framework for understanding machine learning issues such as overfitting and the relationship between successful generalization and the number of training examples considered.

- A *random variable* can be viewed as the name of an experiment with a probabilistic outcome. Its value is the outcome of the experiment.
- A *probability distribution* for a random variable Y specifies the probability $Pr(Y=y_i)$ that Y will take on the value y_i, for each possible value y_i.
- The *expected value*, or *mean*, of a random variable Y is $E[Y] = \sum_i y_i \Pr(Y = y_i)$.
  The symbol μ_Y is commonly used to represent E[Y].
- The *variance* of a random variable is $Var(Y) = E[(Y- \mu_Y)^2]$. The variance characterizes the width or dispersion of the distribution about its mean.
- The *standard deviation* of Y is $\sqrt{Var(Y)}$. The symbol σ_Y is often used to represent it.

- An *estimator* is a random variable Y used to estimate some parameter p of an underlying population.

In our case, when we measure the sample error we are performing an experiment with a random outcome. We first collect a random sample S of n independently drawn instances from the distribution D, and then measure the sample error $error_S(h)$. If we were to repeat this experiment many times, each time drawing a different random sample S_i of size n, we would expect to observe different values for the various $error_S(h)$, depending on random differences in the makeup of the various S_i. We say in such cases that $error_{Si}(h)$, the outcome of the i^{th} such experiment, is a random variable. In general, the value of the random variable is the observed outcome of the random experiment.

Imagine that we were to run k such random experiments, measuring the random variables $error_{S1}(h)$, $error_{S2}(h)$… $error_{Sk}(h)$. Imagine further that we then plotted a histogram displaying the frequency with which we observed each possible error value. As we allowed k to grow, the histogram would approach the form of the distribution shown in the following figure. This figure describes a particular probability distribution called the Binomial distribution. The detailed discussion about Binomial distribution will carry on in the next week.



$$P(r) = \frac{n!}{r!(n-r)!}\, error_{\mathcal{D}}(h)^r (1 - error_{\mathcal{D}}(h))^{n-r} \tag{1.6}$$

References

Websites

1. B-Course
   http://b-course.hiit.fi
2. Tetrad
   http://www.phil.cmu.edu/projects/tetrad.

Books

3. Pearl, Judea; Causality: Models, Reasoning, and Inference, 2000
4. Mitchell, Tom; Machine Learning, 1997 (our textbook)