Data Mining
Classification: Model Evaluation

Lecture Notes for Chapter 4

Introduction to Data Mining

by

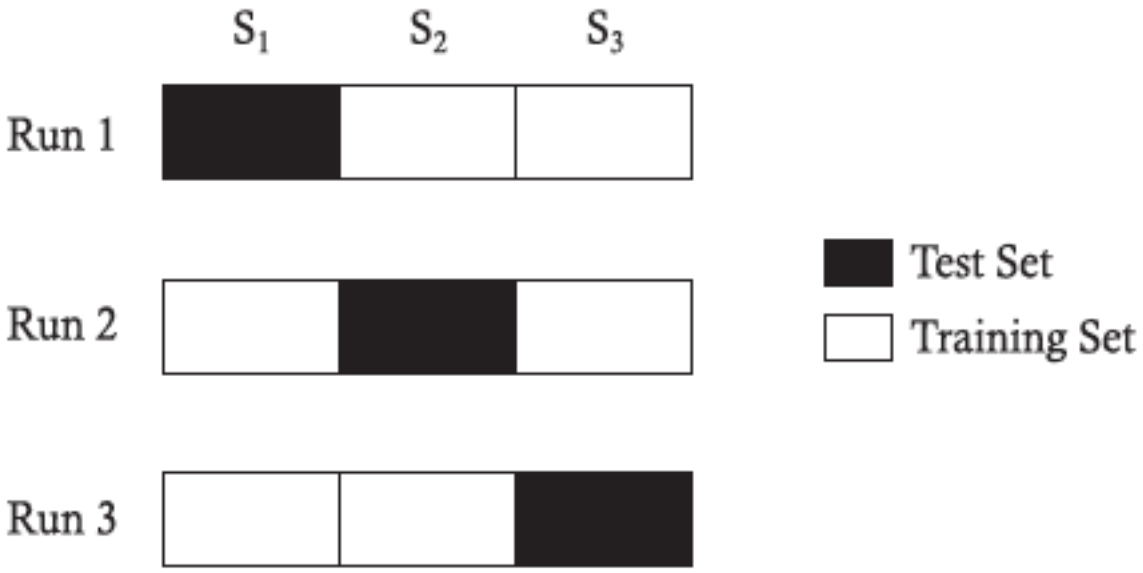
Tan, Steinbach, Kumar

Model Evaluation

- Purpose:
  - To estimate performance of classifier on previously unseen data (test set)
- Holdout
  - Reserve k% for training and (100-k)% for testing
    - Proportion Left at the discretion of the analysts (e.g., 50-50 or two thirds for training and one-third for testing).
  - Random subsampling: repeated holdout
- Cross validation
  - Partition data into k disjoint subsets
  - k-fold: train on k-1 partitions, test on the remaining one
  - Leave-one-out: k=n

Cross-validation Example

- 3-fold cross-validation



Model Evaluation

- Performans Değerlendirmesi için Metrikler
  - **Bir modelin performansı** nasıl değerlendirilir?
- Performans Değerlendirme Yöntemleri
  - **Güvenilir tahminler** nasıl elde edilir?
- Model Karşılaştırma Yöntemleri
  - Rakip modeller arasında **göreceli performans** nasıl karşılaştırılır?

Model Evaluation

- Performans Değerlendirmesi için Metrikler
  - **Bir modelin performansı** nasıl değerlendirilir?
- Performans Değerlendirme Yöntemleri
  - **Güvenilir tahminler** nasıl elde edilir?
- Model Karşılaştırma Yöntemleri
  - Rakip modeller arasında **göreceli performans** nasıl karşılaştırılır?

Metrics for Performance Evaluation

- Bir modelin tahmin yeteneğine (**predictive capability**) odaklanır
  - Sınıflandırma hızı, model oluşturma hızı, ölçeklenebilirik vb hususlardan ziyade…
- Confusion Matrix:

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a	b
	Class=No	c	d

a: TP (true positive)

b: FN (false negative)

c: FP (false positive)

d: TN (true negative)

Metrics for Performance Evaluation…

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

- Most widely-used metric:

$$\text{Accuracy} = \frac{a+d}{a+b+c+d} = \frac{TP+TN}{TP+TN+FP+FN}$$

Metrics for Performance Evaluation…

- **True positive** (TP) or *f*++, sınıflandırma modeli tarafından doğru bir şekilde tahmin edilen pozitif örneklerin sayısına karşılık gelir.
- **False negative** (FN) or *f*+−, sınıflandırma modeli tarafından yanlış bir şekilde negatif olarak tahmin edilen pozitif örneklerin sayısına karşılık gelir.
- **False positive** (FP) or *f*−+, sınıflandırma modeli tarafından yanlış bir şekilde pozitif olarak tahmin edilen negatif örneklerin sayısına karşılık gelir.
- **True negative** (TN) or *f*−−, sınıflandırma modeli tarafından doğru bir şekilde tahmin edilen negatif örneklerin sayısına karşılık gelir.

Metrics for Performance Evaluation…

- Confusion matrisindeki sayılar ayrıca yüzde olarak da ifade edilebilir.

- **True positive rate** (*TPR*) veya **sensitivity (**hassasiyet**),** model tarafından doğru şekilde tahmin edilen pozitif örneklerin oranı olarak tanımlanır, yani

$$TPR = TP/(TP + FN).$$

- **True negative rate** (*TNR*) veya **specificity,** model tarafından doğru bir şekilde tahmin edilen negatif örneklerin oranı olarak tanımlanır, yani,

$$TNR = TN/(TN + FP).$$

- **False positive rate** (*FPR*), pozitif bir sınıf olarak tahmin edilen negatif örneklerin oranıdır, yani

$$FPR = FP/(TN + FP),$$

- **False negative rate** (*FNR*), negatif bir sınıf olarak tahmin edilen pozitif örneklerin oranıdır, yani,

$$FNR = FN/(TP + FN).$$

Limitation of Accuracy

- 2-sınıflı bir problem düşünün
  - Number of Class 0 examples = 9990
  - Number of Class 1 examples = 10

- Model her şeyin *Sınıf-0* olacağını öngörürse, doğruluk 9990/10000 =% 99,9'dur.
  - Doğruluk (**Accuracy**) yanıltıcıdır çünkü model herhangi bir *Sınıf-1* örneği tespit etmez

its limitation is obvious for **imbalanced datasets**

Cost Matrix

		PREDICTED CLASS	
ACTUAL CLASS	C(i|j)	**Class=Yes**	**Class=No**
	Class=Yes	C(Yes|Yes)	C(No|Yes)
	Class=No	C(Yes|No)	C(No|No)

C(i|j): Sınıf j örneğini sınıf i olarak yanlış sınıflandırmanın maliyeti

Computing Cost of Classification

Cost Matrix	PREDICTED CLASS		
	C(i|j)	+	-
ACTUAL CLASS	+	-1	100
	-	1	0

Model M_1	PREDICTED CLASS		
		+	-
ACTUAL CLASS	+	150	40
	-	60	250

Accuracy = 80%

Cost = 3910

Model M_2	PREDICTED CLASS		
		+	-
ACTUAL CLASS	+	250	45
	-	5	200

Accuracy = 90%

Cost = 4255

Cost vs Accuracy

Count	PREDICTED CLASS		
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a	b
	Class=No	c	d

Cost	PREDICTED CLASS		
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	p	q
	Class=No	q	p

Accuracy is proportional to cost if
1. C(Yes|No)=C(No|Yes) = q
2. C(Yes|Yes)=C(No|No) = p

$N = a + b + c + d$

$Accuracy = (a + d)/N$

$$\begin{aligned} Cost &= p\,(a + d) + q\,(b + c) \\ &= p\,(a + d) + q\,(N - a - d) \\ &= q\,N - (q - p)(a + d) \\ &= N\,[q - (q-p) \times Accuracy] \end{aligned}$$

Cost-Sensitive Measures

$$\text{Precision (p)} = \frac{a}{a+c}$$

$$\text{Recall (r)} = \frac{a}{a+b}$$

$$\text{F - measure (F)} = \frac{2rp}{r+p} = \frac{2a}{2a+b+c}$$

In principle, *F-measure* (F_1) represents a harmonic mean between recall and precision, i.e

$$F_1 = \frac{2}{\frac{1}{r} + \frac{1}{p}}$$

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

- **Precision** is biased towards C(Yes|Yes) & C(Yes|No)
- **Recall** is biased towards C(Yes|Yes) & C(No|Yes)
- **F-measure** is biased towards all except C(No|No)

$$\text{Weighted Accuracy} = \frac{w_1 a + w_4 d}{w_1 a + w_2 b + w_3 c + w_4 d}$$

Cost-Sensitive Measures

- **Precision**, sınıflandırıcının pozitif bir sınıf olarak bildirdiği grupta gerçekte pozitif çıkan kayıtların oranını belirler.

- **Recall**, sınıflandırıcı tarafından doğru bir şekilde tahmin edilen pozitif örneklerin oranını ölçer.

Model Evaluation

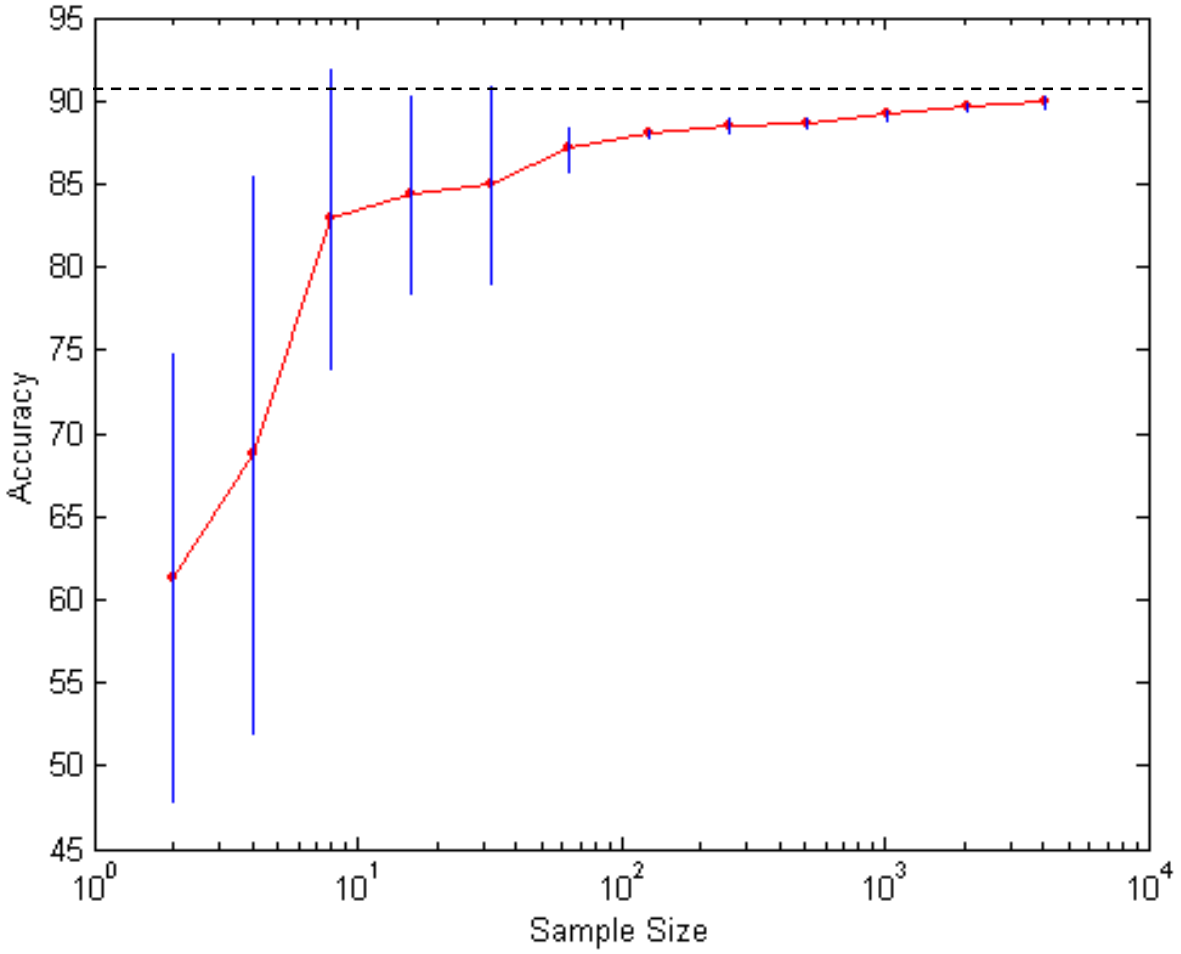
- Performans Değerlendirmesi için Metrikler
  - Bir modelin performansı nasıl değerlendirilir?
- Performans Değerlendirme Yöntemleri
  - Güvenilir tahminler nasıl elde edilir?
- Model Karşılaştırma Yöntemleri
  - Rakip modeller arasında göreceli performans nasıl karşılaştırılır?

Methods for Performance Evaluation

- Güvenilir bir performans tahmini nasıl elde edilir?

- Bir modelin performansı, öğrenme algoritmasının yanı sıra diğer faktörlere de bağlı olabilir:
  - Sınıf dağılımı (*Class distribution*)
  - Yanlış sınıflandırma maliyeti (*Cost of misclassification*)
  - Eğitim ve test setlerinin boyutu

Learning Curve



- Learning curve shows **how accuracy changes with varying sample size**
- Requires a sampling schedule for creating learning curve:
  - Arithmetic sampling (Langley, et al)
  - Geometric sampling (Provost et al)

Effect of small sample size:

- Bias in the estimate
- Variance of estimate

Methods of Estimation

- Holdout
  - Reserve 2/3 for training and 1/3 for testing
- Random subsampling
  - Repeated holdout
- Cross validation
  - Partition data into k disjoint subsets
  - k-fold: train on k-1 partitions, test on the remaining one
  - Leave-one-out: k=n

Model Evaluation

- Performans Değerlendirmesi için Metrikler
  - Bir modelin performansı nasıl değerlendirilir?
- Performans Değerlendirme Yöntemleri
  - Güvenilir tahminler nasıl elde edilir?
- Model Karşılaştırma Yöntemleri
  - Rakip modeller arasında göreceli performans nasıl karşılaştırılır?

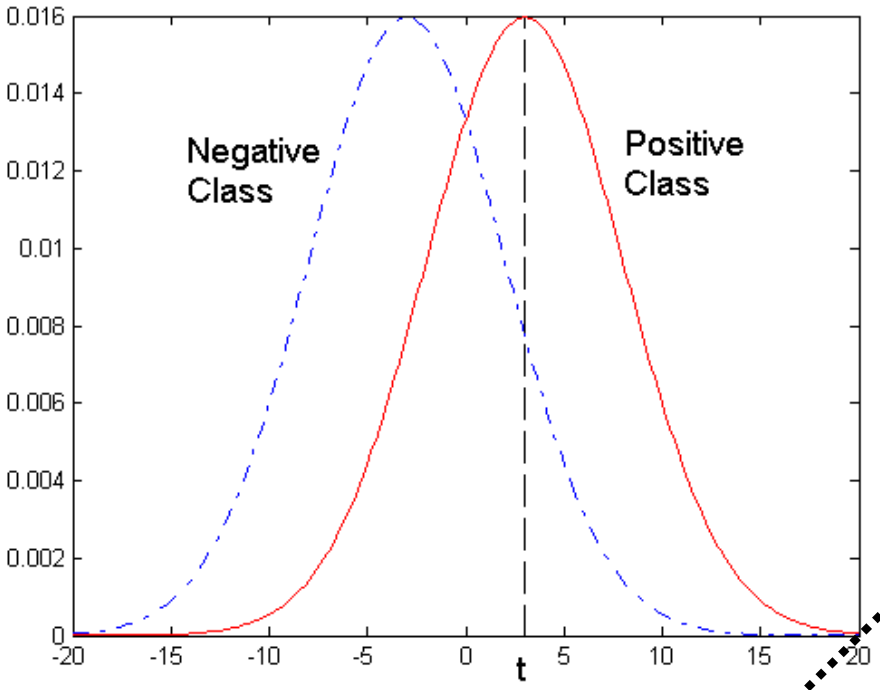
ROC (Receiver Operating Characteristic)

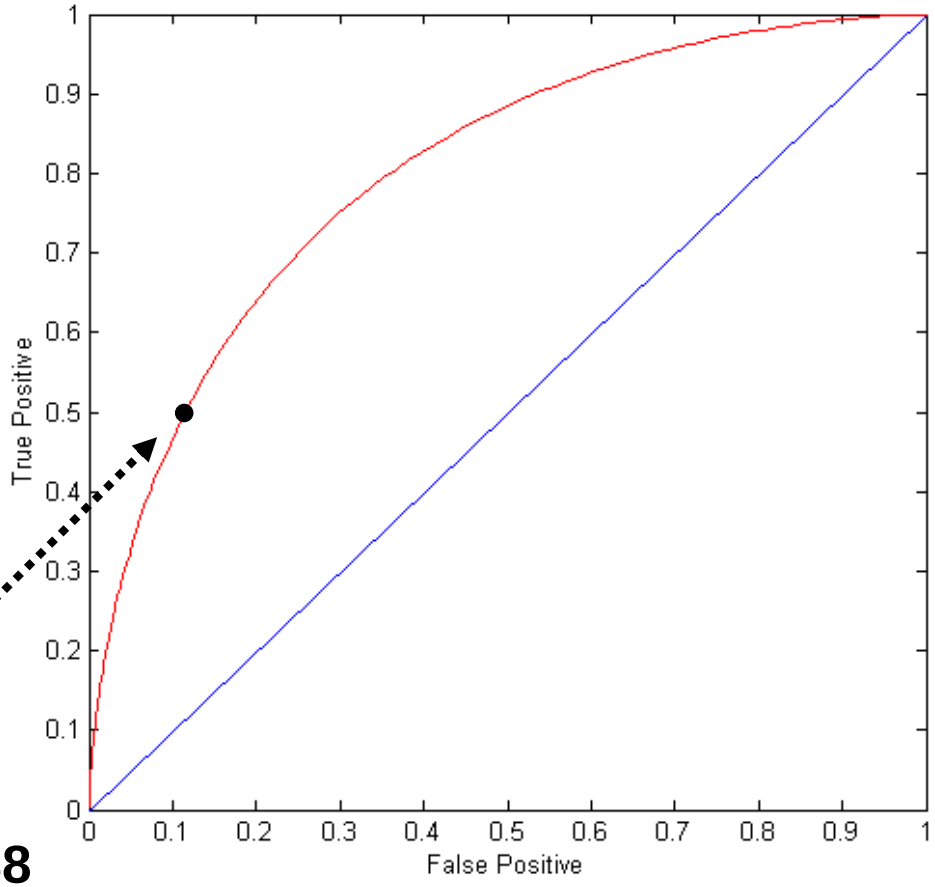
- 1950'lerde gürültülü sinyalleri analiz etmek amacıyla sinyal algılama teorisi için geliştirildi
  - Pozitif isabetler ve yanlış alarmlar arasındaki ödünleşimi karakterize eder (**trade-off between positive hits and false alarms**)
  - ROC eğrisi (curve), TP oranını (y ekseninde) FP oranına (x ekseninde) karşı karakterize eder
- Her sınıflandırıcının performansı ROC eğrisinde bir nokta olarak temsil edilir
  - Algoritmanın eşiğini, örneklem dağılımını veya maliyet matrisini değiştirme noktanın konumunu değiştirir.

ROC Curve

- 1-dimensional data set containing 2 classes (positive and negative)

- any points located at x > t is classified as positive





At threshold t:

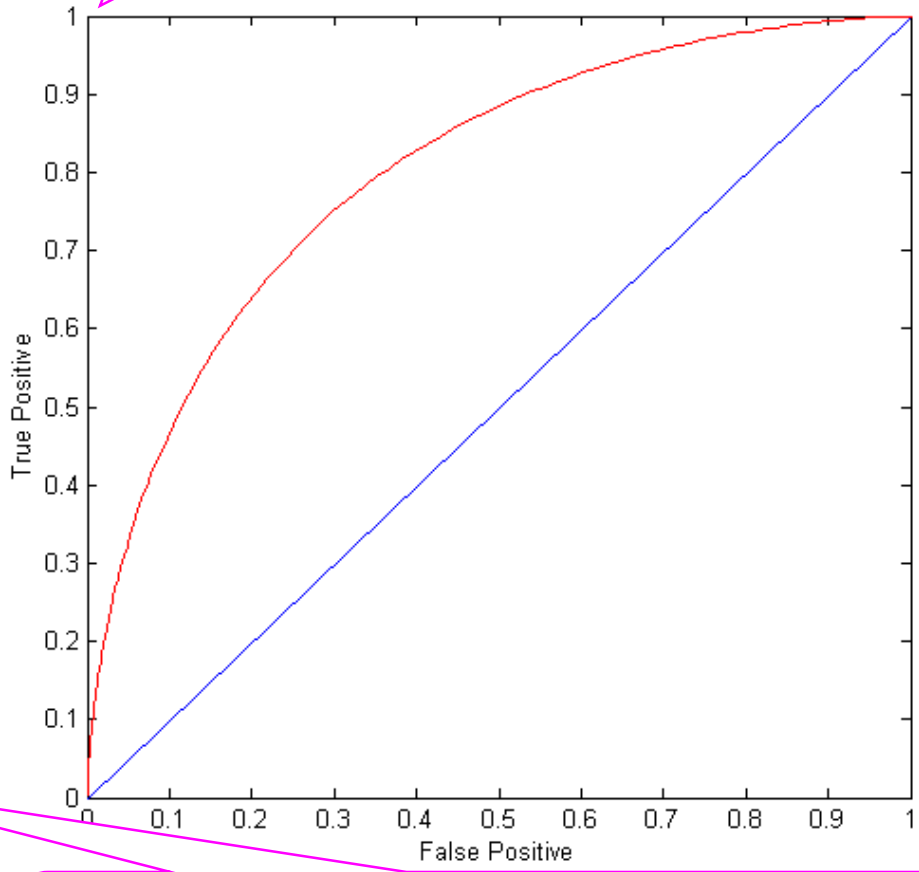
TP=0.5, FN=0.5, FP=0.12, TN=0.88

ROC Curve

İyi bir sınıflandırma modeli, diyagramın sol üst köşesine mümkün olduğunca yakın konumlanmalıdır.

Bir ROC eğrisi boyunca birkaç kritik nokta vardır

- (TPR=0, FPR=0):Model, her örneğin bir negatif sınıf olacağını öngörür.
- (TPR=1, FPR=1):Model, her örneğin pozitif bir sınıf olduğunu öngörür.
- (TPR=1, FPR=0): İdeal model.



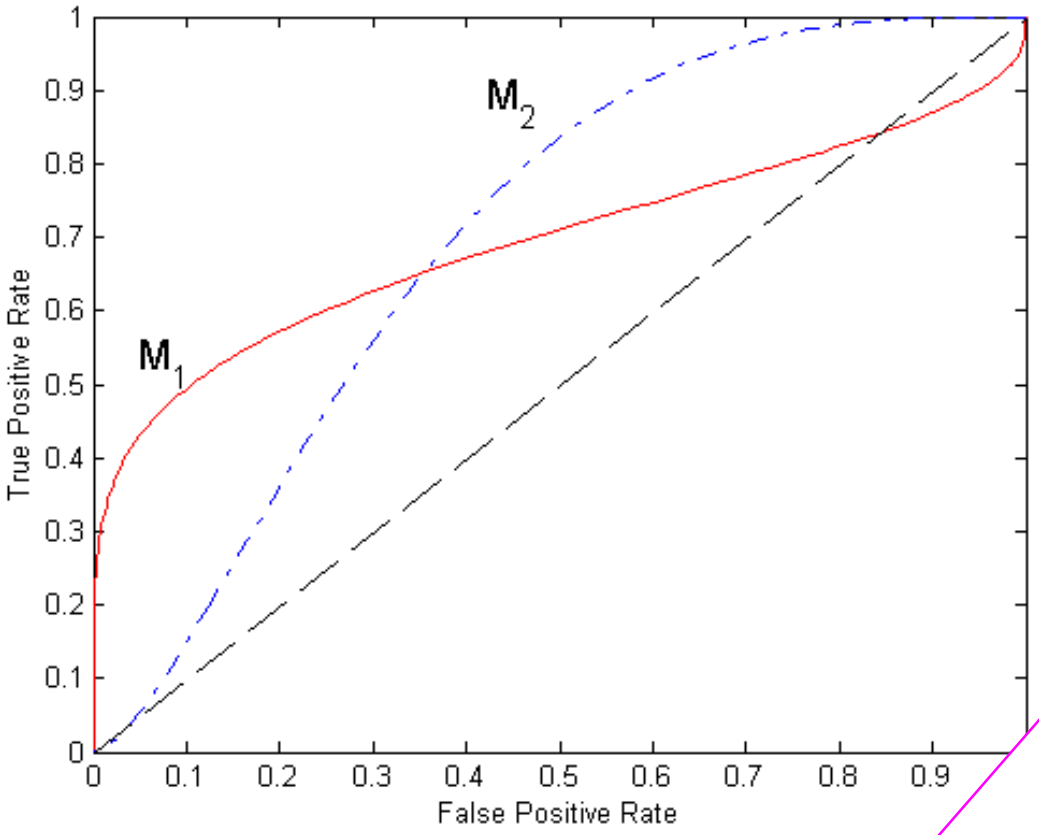
- Köşegen (*Diagonal line*):
  - Random guessing
  - Below diagonal line:
    - prediction is opposite of the true class

Rastgele tahmin (***Random guessing***), bir kaydın, öznitelik kümesine bakılmaksızın, sabit bir olasılık *p* ile pozitif bir sınıf olarak sınıflandırılması anlamına gelir.

Rastgele tahmin, bir kaydın, öznitelik kümesine bakılmaksızın, sabit bir olasılık p ile pozitif bir sınıf olarak sınıflandırılması anlamına gelir.

- Örneğin, n_+ pozitif örnekler ve n_- negatif örnekler içeren bir veri kümesi düşünün.
- Rastgele sınıflandırıcının pozitif örneklerin pn_+ 'sini doğru şekilde sınıflandırması ve negatif örneklerin pn_- 'sini yanlış sınıflandırması beklenir.
- Bu nedenle, sınıflandırıcının TPR'si $(pn_+)/n_+ = p$, FPR'si $(pn_-)/n_- = p$.
- TPR ve FPR aynı olduğundan, **rastgele sınıflandırıcı için ROC eğrisi** her zaman ana köşegen boyunca yer alır.

Using ROC for Model Comparison



- (In this example) No model consistently outperform the other
  - M_1 is better for small FPR
  - M_2 is better for large FPR
- Area Under the ROC curve
  - Ideal:
    - Area = 1
  - Random guess:
    - Area = 0.5

The area under the ROC curve (AUC) provides another approach for evaluating **which model is better on average**.

How to Construct an ROC curve

Instance	P(+|A)	True Class
1	0.95	+
2	0.93	+
3	0.87	-
4	0.85	-
5	0.85	-
6	0.85	+
7	0.76	-
8	0.53	+
9	0.43	-
10	0.25	+

- Use classifier that produces posterior probability for each test instance P(+|A)
- Sort the instances according to P(+|A) in decreasing order
- Apply threshold at each unique value of P(+|A)
- Count the number of TP, FP, TN, FN at each threshold
- TP rate, TPR = TP/(TP+FN)
- FP rate, FPR = FP/(FP + TN)

How to Construct an ROC curve

1. Sürekli değerli çıktıların pozitif sınıf için tanımlandığını varsayarak, kayıtları çıktı değerlerinin artan sırasına göre sıralayın.

2. En düşük dereceli test kaydını seçin (yani, en düşük çıktı değerine sahip kaydı). Seçilen kaydı ve üzerinde sıralananları pozitif sınıfa atayın. Bu yaklaşım, tüm test kayıtlarının pozitif sınıf olarak sınıflandırılmasına eşdeğerdir. Tüm pozitif örnekler doğru şekilde sınıflandırıldığı ve negatif örnekler yanlış sınıflandırıldığı için, TPR = FPR = 1.

3. Sıralanan listeden sonraki test kaydını seçin. Seçili kaydı ve üzerinde sıralananları pozitif, altında olanları negatif olarak sınıflandırın. Önceden seçilen kaydın gerçek sınıf etiketini inceleyerek TP ve FP sayılarını güncelleyin. Önceden seçilen kayıt pozitif bir sınıfsa, TP sayısı azaltılır ve FP sayısı öncekiyle aynı kalır. Önceden seçilen kayıt negatif bir sınıfsa, FP sayısı azaltılır ve TP sayısı öncekiyle aynı kalır.

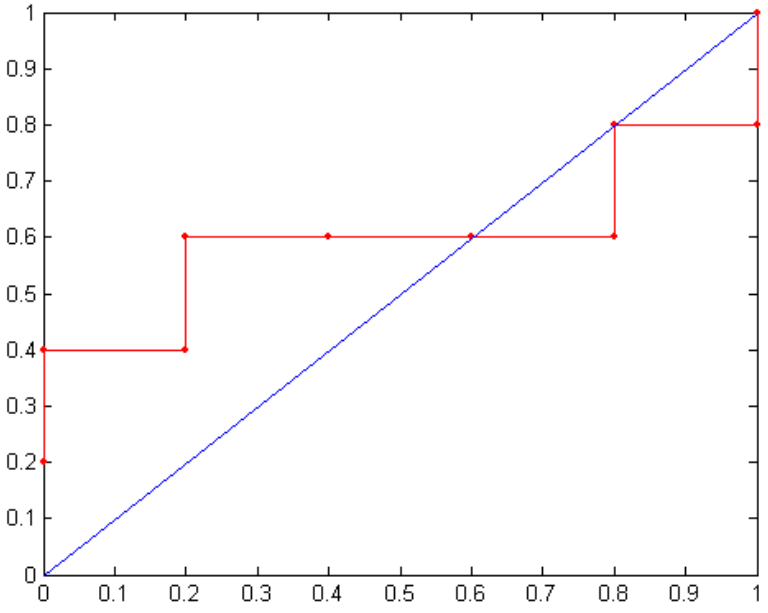
4. Adımı tekrarlayın ve en yüksek dereceli test kaydı seçilene kadar TP ve FP sayılarını uygun şekilde güncelleyin.

5. Sınıflandırıcının FPR'sine karşı TPR'yi çizin.

How to construct an ROC curve

	Class	+	-	+	-	-	-	+	-	+	+	
Threshold >=		0.25	0.43	0.53	0.76	0.85	0.85	0.85	0.87	0.93	0.95	1.00
	TP	5	4	4	3	3	3	3	2	2	1	0
	FP	5	5	4	4	3	2	1	1	0	0	0
	TN	0	0	1	1	2	3	4	4	5	5	5
	FN	0	1	1	2	2	2	2	3	3	4	5
→	TPR	1	0.8	0.8	0.6	0.6	0.6	0.6	0.4	0.4	0.2	0
→	FPR	1	1	0.8	0.8	0.6	0.4	0.2	0.2	0	0	0

ROC Curve:



Test of Significance

- Given two models:
  - Model M1: accuracy = 85%, tested on 30 instances
  - Model M2: accuracy = 75%, tested on 5000 instances

- M1'in M2'den daha iyi olduğunu söyleyebilir miyiz?
  - M1 ve M2'nin doğruluğuna ne kadar güvenebiliriz?
  - Performans ölçüsündeki fark, **test setindeki rastgele dalgalanmaların** bir sonucu olarak açıklanabilir mi?

Confidence Interval for Accuracy

- Güven aralığını belirlemek için, **doğruluk (accuracy) ölçüsünü yöneten olasılık dağılımını oluşturmamız gerekir**.
- Sınıflandırma görevini binom deneyi olarak modelleyerek güven aralığını türetmek için bir yaklaşıma ihtiyacımız var.
- Aşağıda bir binom deneyinin (Binomial Experiment) özelliklerinin bir listesi verilmiştir:
  1. Deney, her denemenin iki olası sonuca sahip olduğu *N* bağımsız denemeden oluşur: başarı (**success**) veya başarısızlık (**failure**).
  2. Her denemede başarı olasılığı, *p*, sabittir.

Confidence Interval for Accuracy

- **Binom deneyine bir örnek**, (bir yazı tura denemsinde) bozuk para *N* kez atıldığında ortaya çıkan tura sayısını saymaktır.
- *X*, *N* denemede gözlemlenen başarı sayısı ise, ***X*'in belirli bir değeri alma olasılığı**, ortalama *Np* ve varyans *Np* (1 - *p*) olan bir binom dağılımı ile verilir:

$$P(X = v) = \binom{N}{v} p^v (1-p)^{N-v}.$$

- Örneğin, bozuk para adil (*fair coin*) ise (*p* = 0.5) ve elli kez atılmışsa, turanın 20 kez ortaya çıkma olasılığı

$$P(X = 20) = \binom{50}{20} 0.5^{20} (1-0.5)^{30} = 0.0419.$$

- Deney birçok kez tekrarlanırsa, ortaya çıkması beklenen ortalama tura sayısı 50 × 0.5 = 25 iken varyansı 50 × 0.5 × 0.5 = 12.5'tir.

Confidence Interval for Accuracy

- Tahmin, bir Bernoulli denemesi olarak kabul edilebilir
  - Bernoulli denemesinin 2 olası sonucu vardır
  - Tahmin için olası sonuçlar: doğru veya yanlış
  - Bernoulli denemeleri koleksiyonunun Binom dağılımı vardır:
    - $x \sim Bin(N, p)$ x: doğru tahmin sayısı
    - e.g: Adil bir bozuk parayı 50 kez atarsan, kaç tura çıkar?
      Beklenen tura sayısı = $N \times p = 50 \times 0.5 = 25$

- X (doğru tahmin sayısı) verildiğinde veya eşdeğer olarak, acc = x / N ve N (test örneği sayısı) verildiğinde,

  p'yi (modelin gerçek doğruluğunu) tahmin edebilir miyiz?

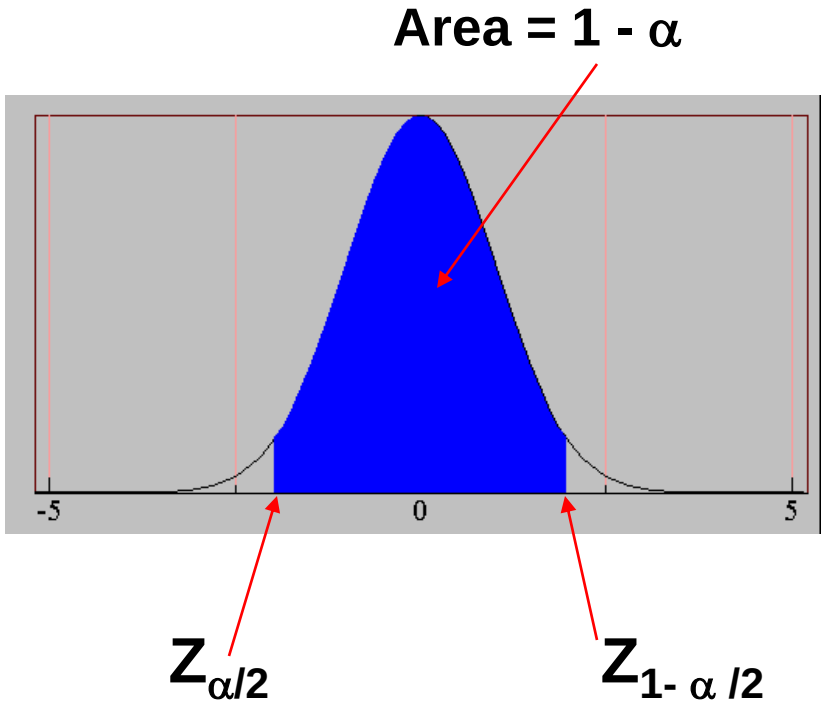
true accuracy of model

Confidence Interval for Accuracy

- Test kayıtlarının sınıf etiketlerini tahmin etme görevi de bir binom deneyi (**binomial experiment**) olarak düşünülebilir.
- *N* kayıt içeren bir test seti verildiğinde, *X* bir model tarafından **doğru tahmin edilen kayıt sayısı** ve *p* modelin gerçek doğruluğu (the true accuracy) olsun.
- Tahmin görevini binom deneyi olarak modelleyerek, *X*; *Np* ortalama ve *Np* (1 - *p*) varyans ile bir binom dağılımına sahiptir.
- Deneysel doğruluğun, acc = *X* / *N*, aynı zamanda *p* ortalama ve *p* (1 − *p*) / *N* varyans ile bir binom dağılımına sahip olduğu gösterilebilir (bkz. önceki slaytlar).
- Binom dağılım, *acc* için güven aralığını tahmin etmek amacıyla kullanılabilmesine rağmen, *N* **yeterince büyük** olduğunda genellikle normal dağılımla yaklaşık olarak tahmin edilir.

Confidence Interval for Accuracy

- For large test sets (N > 30),
  - acc has a normal distribution with mean p and variance p(1-p)/N



$$P(Z_{\alpha/2} < \frac{acc - p}{\sqrt{p(1-p)/N}} < Z_{1-\alpha/2}) = 1 - \alpha$$

- Confidence Interval for p:

$$p = \frac{2 \times N \times acc + Z_{\alpha/2}^2 \pm \sqrt{Z_{\alpha/2}^2 + 4 \times N \times acc - 4 \times N \times acc^2}}{2(N + Z_{\alpha/2}^2)}$$

Confidence Interval for Accuracy

- Consider a model that produces an accuracy of 80% when evaluated on 100 test instances:
  - N=100, acc = 0.8
  - Let 1-α = 0.95 (95% confidence)
  - From probability table, $Z_{\alpha/2}$=1.96

1-α	Z
0.99	2.58
0.98	2.33
0.95	1.96
0.90	1.65

N	50	100	500	1000	5000
p(lower)	0.670	0.711	0.763	0.774	0.789
p(upper)	0.888	0.866	0.833	0.824	0.811

Note that the confidence interval becomes tighter when *N* increases

Confidence Interval: 71.1% and 86.7%

Comparing Performance of 2 Models

- Given two models, say M1 and M2, which is better?
  - M1 is tested on D1 (size=n1), found error rate = e_1
  - M2 is tested on D2 (size=n2), found error rate = e_2
  - Assume D1 and D2 are independent test sets
  - If n1 and n2 are sufficiently large, then

$$e_1 \sim N(\mu_1, \sigma_1)$$
$$e_2 \sim N(\mu_2, \sigma_2)$$

the error rates e_1 and e_2 can be approximated using normal distributions.

  - Approximate: $\hat{\sigma}_i = \frac{e_i(1-e_i)}{n_i}$

Comparing Performance of 2 Models

- To test if performance difference is **statistically significant**: d = e1 – e2
  - d ~ $N(d_t,\sigma_t)$ where $\mathbf{d_t}$ is the **true difference**
  - Since D1 and D2 are independent, their variance adds up:

$$\sigma_d^2 = \sigma_1^2 + \sigma_2^2 \cong \hat{\sigma}_1^2 + \hat{\sigma}_2^2$$

$$= \frac{e1(1-e1)}{n1} + \frac{e2(1-e2)}{n2}$$

Our goal is to test whether the observed difference between *e*1 and *e*2 is statistically significant.

If the observed difference in the error rate is denoted as $d = e1 - e2$, then d is also normally distributed with mean $\boldsymbol{d_t}$, its true difference, and variance, σ_d^2

it can be shown that the **confidence interval for the true difference d_t** is given by this equation

  - At (1-α) confidence level, $d_t = d \pm Z_{\alpha/2}\hat{\sigma}_d$

An Illustrative Example

- Given: M1: n1 = 30, e1 = 0.15
  M2: n2 = 5000, e2 = 0.25
- d = |e2 – e1| = 0.1 (2-sided test)

In this example, we are performing a two-sided test to **check whether *dt* = 0 or *dt* ≠0**.

$$\hat{\sigma}_d^2 = \frac{0.15(1-0.15)}{30} + \frac{0.25(1-0.25)}{5000} = 0.0043$$

Estimated variance

- At 95% confidence level, $Z_{\alpha/2}$=1.96

$$d_t = 0.100 \pm 1.96 \times \sqrt{0.0043} = 0.100 \pm 0.128$$

=> Interval contains 0 => difference may not be statistically significant

As the interval spans the value zero, we can conclude that the observed difference is not **statistically significant** at a 95% confidence level.

Comparing Performance of 2 Algorithms (Classifiers)

- Each learning algorithm may produce k models:
  - L1 may produce M11 , M12, …, M1k
  - L2 may produce M21 , M22, …, M2k
- If models are generated on the same test sets D1,D2, …, Dk (e.g., via cross-validation)
  - For each set: compute $d_j = e_{1j} - e_{2j}$
  - d_j has mean d_t and variance σ_t
  - Estimate:

$$\hat{\sigma}_t^2 = \frac{\sum_{j=1}^{k} (d_j - \bar{d})^2}{k(k-1)}$$

$$d_t = d \pm t_{1-\alpha, k-1} \hat{\sigma}_t$$