

Strojno učenje

3 (I dio)

Evaluacija modela

Tomislav Šmuc

PMF, Zagreb, 2020

- i. Greške (stvarna; T - na osnovu uzorka primjera)
- ii. Resampling metode procjene greške
- iii. Usporedba modela ili algoritama (na istim podacima)
- iv. Mjere
 - i. Klasifikacija:
 - Krivulja učenja, TP,FP... Matrica konfuzije
 - točnost, osjetljivost, preciznost
 - ii. IR (+ klasifikacija): F_1 , ROC (AUC)

Tokom drugih predavanja – uz pojedina područja

- i. Regresija – RMSE; RAE
- ii. Clustering: Mjere “dobrote” clusteringa
- iii. Učenje pravila (podrška/pouzdanost/”pojačanje” - support/confidence/lift)

Structural Risk Minimization i VC dimenzija

Pretpostavimo da imamo na izbor niz “strojeva” ili algoritama
– koji uče hipoteze iz prostora H_i (funkcije) različitih $VC(H_i)$ tako da vrijedi:

$$VC(H_1) \leq VC(H_2) \leq VC(H_3) \leq VC(H_4) \leq \dots \leq VC(H_N)$$

Koji ćemo od “strojeva” - algoritama koristiti ?

- Treniramo svaki od strojeva i mjerimo e_T ... i procjenjujemo e_{test} na osnovu:

$$e_{\Delta} \approx e_{test} \leq e_T + \sqrt{\frac{VC(H)(\log(2N / VC(H)) + 1) - \log(\eta / 4)}{N}}$$

rbr	H_i	e_T	$\sqrt{VC(H) \dots}$	$\sim e_{test}$	Rang
1	H_1	<div><div></div></div>	<div><div></div></div>	<div><div></div><div></div></div>	4
2	H_2	<div><div></div></div>	<div><div></div></div>	<div><div></div><div></div></div>	1
3	H_3	<div><div></div></div>	<div><div></div></div>	<div><div></div><div></div></div>	1
4	H_4	<div><div></div></div>	<div><div></div></div>	<div><div></div><div></div></div>	1
5	H_5	<div><div></div></div>	<div><div></div></div>	<div><div></div><div></div></div>	5

Druge metode procjene

bazirane samo na $e_T(h)$ i procjeni rizika

AIC (Akaike Information Criterion)

$$AIC(f(x, \alpha)) = e_T(f(x, \alpha)) + 2 \cdot \frac{d(\alpha)}{N} \cdot \hat{\sigma}_\epsilon^2$$

$d(\alpha)$ – broj parametara modela
 σ - procijenjena pristranost (bias) modela












BIC (Bayesian information Criterion)

$$BIC(f(x, \alpha)) = \frac{N}{2} \left[e_T(f(x, \alpha)) + (\log N) \cdot \frac{d(\alpha)}{N} \cdot \hat{\sigma}_\epsilon^2 \right]$$

MDL (Minimum Description Length)

$$DL = -\log P(\mathbf{y} \mid \alpha, f, \mathbf{x}) - \log P(\alpha \mid h)$$

The Elements of Statistical Learning,
Hastie, Tibshirani, Friedman

rbr	H _i	e _T	AIC/BIC/DL	~e _{test}	Rang
1	H ₁				
2	H ₂				
3	H ₃				
4	H ₄				
5	H ₅				

Glasači vs. Ne-glasači (HR) – binarni klasifikacijski problem

(Skup podataka za učenje – u tekstu *T* ili *S*)

- Godine: ('18-25','26-35','36-45',...,76+)
 - Spol: {M, Ž}
 - Brak: {Da, Ne}
 - Obrazovanje: {nš,oš,sš,vš,všš}
 - Broj djece: ('0','<=2','3+')
 - Regija: {I,S,J,Z,C}
- Primanja {<50, 50-100, 100-200, >200}
 - Zaduženost (kredit): {0-50, 50-100, 100-200, >200}
 - Najčešće čita novine {V, JL, M, SN, Os}
 - Klasa: {1,0}

Godine	Spol	Brak	Obrazovanje	Broj djece	Regija	Primanja (kHRK/god)	Zaduženost (kHRK)	Novine	Klasa G(+)/ NG(-)
'26-35'	m	Da	sš	'<=2'	I	'50-100'	'50-100'	V	-
'26-35'	ž	Ne	vss	'0'	S	'<50'	'0-50'	JL	+
'56-65'	ž	Da	ss	'3+'	J	'100-200'	'100-200'	Os	-
'66-75'	m	Ne	vss	'<=2'	Z	'<50'	'>100'	M	+
'18-25'	m	Ne	ss	'0'	C	'<50'	'0-50'	SN	+
.....

X

Y

Kako mjerimo grešku: Empirijska evaluacija uspješnosti učenja

Neki (očiti) zaključci:

- a. $e_T(h)$ je gotovo uvijek pristrana (en biased) /optimistična procjena $e_{\Delta}(h)$

$$(\text{bias} \equiv E[e_T(h)] - e_{\Delta}(h))$$

Da izbjegnemo ovaj optimistični *bias*, h i T moraju biti odabrani nezavisno !?

- a. Čak i ako imamo ne-pristrano odabran skup točaka T , $e_T(h)$ se može značajno razlikovati od $e_{\Delta}(h)$

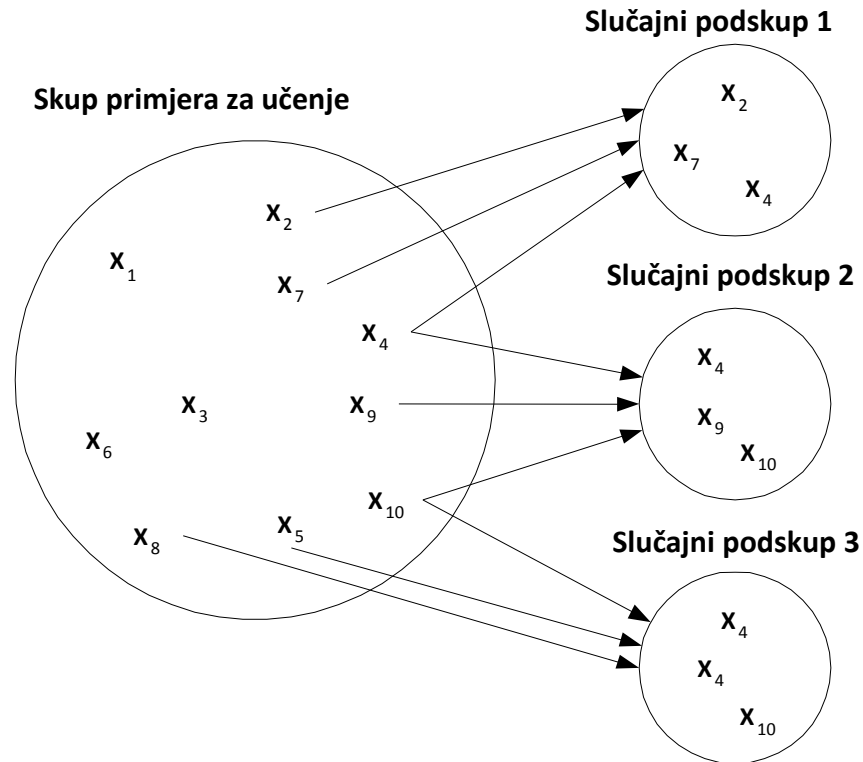
$$\text{Varijanca } X \Rightarrow V(X) \equiv E[(X - E(X))^2]$$

Empirijska procjena greške - validacija modela

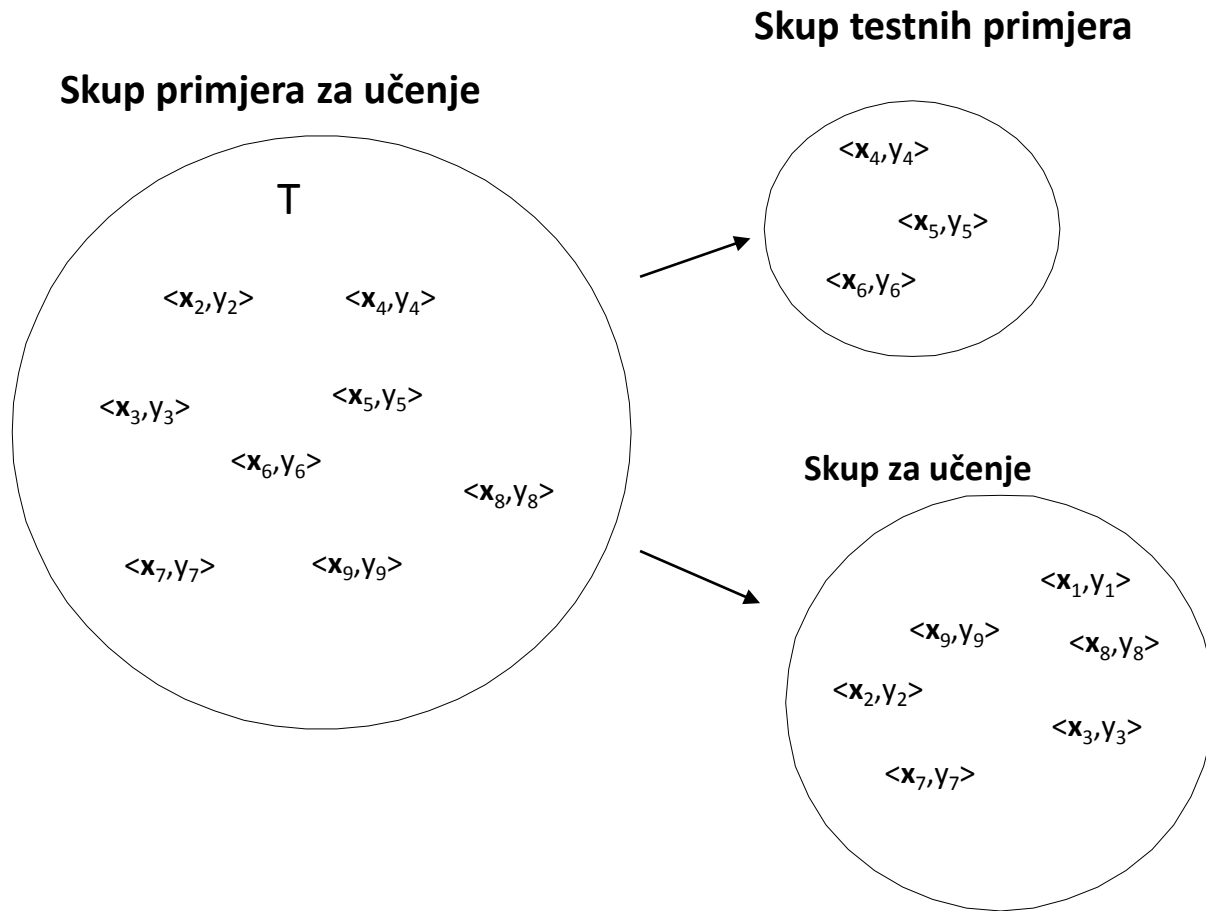
Tehnike probira (en. resampling)

- ☐ Train & Test metoda
- ☐ Unakrsna validacija (en. Cross-Validation)
- ☐ LOOCV (en. Leave-One-Out-Cross-Validation)
 - ☐ Bootstrap sampling (Out-of-Bag error)

Probir - osnove



Probir - osnove



Train & test metoda

- a. Slučajno odaberemo $1/3$ od dostupnih primjera za učenje i stavimo ih u novi – **skup za testiranje (en. Test set)**
- b. Ostatak od $2/3$ primjera iskoristimo za učenje modela – **skup za učenje (en. Training set)**
- c. Naučimo model na skupu za učenje
- d. Procijenimo stvarnu grešku modela “testirajući” novi model na **skupu za testiranje**.

Train & test metoda

Karakteristike

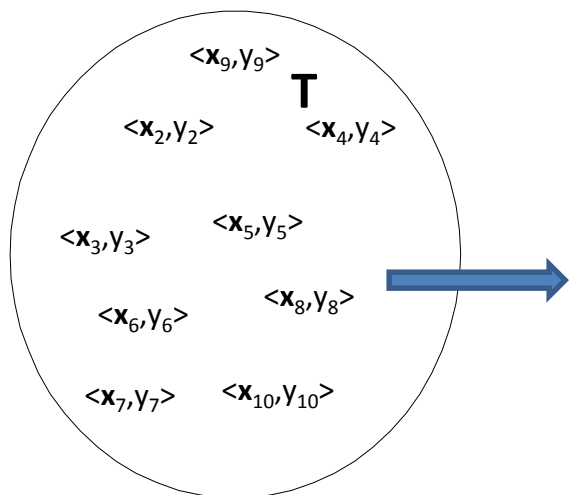
DOBRO:

Jednostavna metoda - odabiremo onaj model koji daje najmanju grešku na testnom skupu

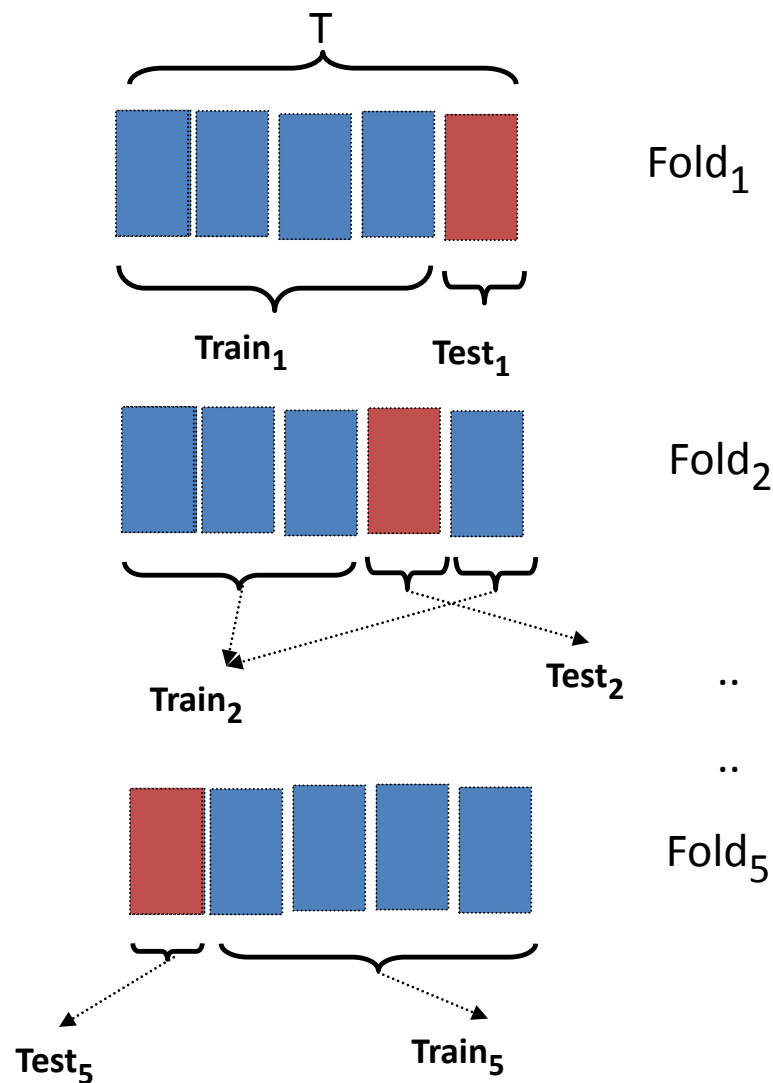
LOŠE:

- a. “Gubimo” vrijedne podatke! 1/3 podataka se uopće ne koristi za izradu modela
- b. Ako imamo relativno malo podataka za učenje ocjena greške na testnom skupu će biti vrlo nepouzdana (>> varijanca greške)

Skup primjera za učenje



Unakrsna validacija (primjer: 5-fold CV)



k-struka unakrsna validacija (k-fold cross validation)

- a. Slučajno rasporediti primjere za učenje u k odvojenih skupova T_i , $i=1, k$ (tipično po 30+ primjera)
- b. Za $i=1$ do k
 - a. Koristi T_i kao testni skup a ostale podatke (T_m $m \neq i$), iskoristi za učenje modela h_i
 - b. Na testnom skupu T_i izračunaj grešku L_m modela h_m
- c. Izračunaj prosječnu grešku za svih k modela

$$\bar{L}_{k-fold} = \frac{1}{k} \sum_{m=1, k} L_m$$

Pojedinačna unakrsna validacija Leave-one-out cross validation (LOOCV)

Na skupu primjera za učenje $(\mathbf{x}_i, y_i) \in D, \quad i=1, N$

- a. Za $i=1$ do N
 - a. Privremeno izdvoji primjer (\mathbf{x}_i, y_i) iz skupa primjera za učenje
 - b. Nauči model h_m na preostalim primjerima $(N-1)$
 - c. Izračunaj grešku modela h_m primjeru (\mathbf{x}_i, y_i)
- b. Izračunaj prosječnu grešku za svih N modela

$$\bar{L}_{LOOCV} = \frac{1}{N} \sum_{m=1, N} L_m$$

Karakteristike metoda evaluacije probirom

Metoda	Dobre strane	Loše strane
Train & Test	<ul style="list-style-type: none"> ■ Jeftina – učimo samo jednom 	<ul style="list-style-type: none"> ■ Gubimo puno primjera za učenje ■ Nepouzdana procjena stvarne greške
K-fold CV	<ul style="list-style-type: none"> ■ Gubimo samo N/k za učenje jednog modela ■ Stabilnija procjena greške 	<ul style="list-style-type: none"> ■ K puta “skuplja” od T&T – učimo k modela
LOOCV	<ul style="list-style-type: none"> ■ Praktički učimo na svim primjerima (-1) ■ Dobra za mali broj primjera 	<ul style="list-style-type: none"> ■ Vrlo skupa za veliki N – učimo N modela !

Statistička evaluacija greške

Statistički problem – određivanje parametara i testiranje hipoteza

Neka je $f(\mathbf{x})$ ciljna funkcija koju želimo naučiti koja savršeno klasificira primjere iz Δ
 Stvarna greška našeg modela

$$e_{\Delta}(h) \equiv P_{\mathbf{x} \in \Delta} [f(\mathbf{x}) \neq h(\mathbf{x})]$$

Ono što možemo lako dobiti jest greška na skupu $S(=T)$ primjera na kojem učimo:

$$e_S(h) \equiv \frac{1}{n} \sum_{\mathbf{x} \in S} \delta(f(\mathbf{x}) \neq h(\mathbf{x})); \quad \delta(f(\mathbf{x}) \neq h(\mathbf{x})) = 1, \quad \delta(f(\mathbf{x}) = h(\mathbf{x})) = 0$$

$e_S(h)$ je rezultat slučajnog eksperimenta (slično je i s e_S^{CV})

Koliko je $e_S(h) / e_S^{CV}$ dobra procjena $e_{\Delta}(h)$?

Statistička evaluacija greške

Primjer:

- a. h griješi na 20 od 100 primjera
- b. $e_s(h) = 20/100 = 0.2$
- c. Koliki je $e_{\Delta}(h)$?

Kao da imamo binarni klasifikator (0,1) - slično bacanju novčića
i neka je $e_{\Delta}(h) = \Theta$ stvarna greška h

Tada imamo u pozadini određivanja $e_{\Delta}(h)$ iz rezultata $e_s(h)$
binomnu distribuciju !

Statistička evaluacija greške

Što predstavlja naš eksperiment ?

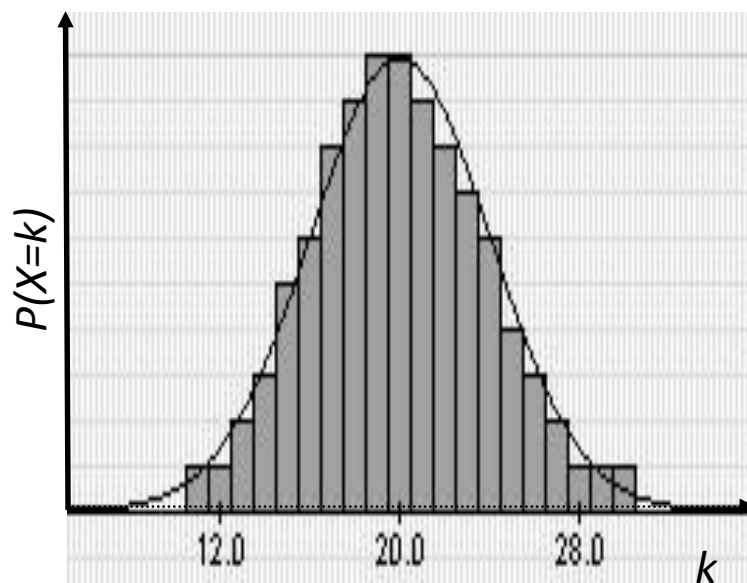
- Neka su h i $e_{\Delta}(h)$ fiksni (poznati)
- S je slučajnog karaktera - eksperiment je dakle probir primjera iz Δ u S !
- $R = e_S(h) \cdot |S|$ - greška je slučajna varijabla koja ovisi o S !

U našem slučaju:

- $|S| = 100$, a $e_S(h) = 0.2$. Koliko je vjerojatno da je ustvari $e_{\Delta}(h) = 0.3$?

Treba pogledati binomnu raspodjelu !

Binomna raspodjela



$P(X=k)$ – vjerojatnost da ćemo imati k puta
ishod=*glava* u n pokušaja (p/g)

$P=P(\text{glava})$

Srednja vrijednost

$$E(X) \equiv \sum_{i=0}^n iP(i) = np$$

Varijanca - $\text{Var}(X) = \sigma^2$

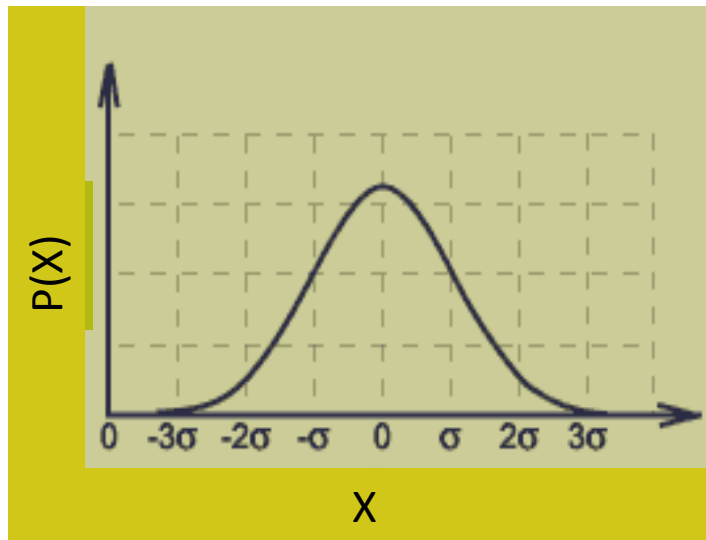
$$\text{Var}(X) \equiv E[(X - E[X])^2] = np(1 - p)$$

$$P(X = k) = \frac{n!}{k!(n - k)!} p^k (1 - p)^{n - k}$$

Standardna devijacija X - $\sigma_X = \sigma$

$$\sigma_X \equiv \sqrt{E[(X - E[X])^2]} = \sqrt{np(1 - p)}$$

Normalna raspodjela



Vjerojatnost da će X biti u intervalu (a,b) je

$$P(X) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

- srednja vrijednost
- Varijanca
- Standardna devijacija

$$E(X)=\mu$$

$$\text{Var}(X)=\sigma^2$$

$$\sigma_X = \sigma$$

Mjerenje uspješnosti učenja

Sve naše pretpostavke:

- h i S su nezavisno odabrani
- $n > 30$ – binomna raspodjela dobro je aproksimirana normalnom raspodjelom
- $\mu(e_S(h)) = e_{\Delta}(h)$

$$\sigma(e_S(h)) = \sqrt{\frac{e_S(h)(1-e_S(h))}{n}} \approx \sqrt{\frac{e_{\Delta}(h)(1-e_{\Delta}(h))}{n}}$$

Za veći n ($n > 30$) normalna raspodjela je dobra aproksimacija binomne raspodjele !

Poznato: Sa $N\%$ vjerojatnosti, $e_{\Delta}(h)$ se nalazi u intervalu:

$$e_{\Delta}(h) = e_s(h) \pm z_N \sqrt{\frac{e_s(h)(1 - e_s(h))}{n}}$$

Gdje vrijedi:

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

Računanje razlika između modela

1. Želimo odrediti razliku u uspješnosti dva modela h_1 i h_2 :

$$\delta \equiv e_{\Delta}(h_1) - e_{\Delta}(h_2)$$

2. Na osnovu procjena dobivenih na S_1 i S_2

$$\hat{\delta} \equiv e_{S_1}(h_1) - e_{S_2}(h_2)$$

3. Odredimo distribuciju vjerojatnosti koja je u pozadini naše procjene

$$\sigma_{\hat{\delta}} \approx \sqrt{\frac{e_{S_1}(h_1)(1 - e_{S_1}(h_1))}{n_1} + \frac{e_{S_2}(h_2)(1 - e_{S_2}(h_2))}{n_2}}$$

4. Na kraju nađemo interval $N\%$ vjerojatnosti u koji spada δ

$$\delta \approx \hat{\delta} \pm z_N \sqrt{\frac{e_{S_1}(h_1)(1 - e_{S_1}(h_1))}{n_1} + \frac{e_{S_2}(h_2)(1 - e_{S_2}(h_2))}{n_2}}$$

Usporedba 2 algoritma strojnog učenja

Vrlo često:

- a. Želimo pronaći najbolji algoritam za naš problem
- b. Odrediti razliku u uspješnosti algoritama i ustanoviti da li je ona statistički značajna

Statistički test mora kontrolirati nekoliko izvora varijacije:

- u izboru testnih podataka
- u izboru podataka za učenje
- slučajni odabiri/odluke u algoritmima

Usporedba dva algoritma (1)

1) Koristeći Trening + test skup i dva algoritma (modela) - L_1 i L_2

$$H_0: \mu_0 = \mu_1 \text{ vs } H_1: \mu_0 \neq \mu_1$$

McNemar test

e_{00} : broj primjera koji su pogrešno klasificirani od strane L_1 i L_2	e_{01} : broj primjera koji su pogrešno klasificirani od strane L_1 ali ne i od L_2
e_{10} : broj primjera koji su pogrešno klasificirani od strane L_2 ali ne i od L_1	e_{11} : broj primjera ispravno klasificiranih od L_1 i L_2

2) Pod H_0 očekujemo:

$$e_{01} = e_{10} = (e_{01} + e_{10})/2$$

$$\frac{(|e_{01} - e_{10}| - 1)^2}{e_{01} + e_{10}} \sim \chi_1^2$$

H_0 je prihvatljiva ako:

$$\frac{(|e_{01} - e_{10}| - 1)^2}{e_{01} + e_{10}} < \chi_{\alpha,1}^2 \quad \alpha \ll 1 \text{ (npr. 0.01)}$$

Usporedba dva algoritma (2)

- k-struki test koristeći unakrsnu validaciju (paired t-test)

p_i^1, p_i^2 : greške algoritma 1 i 2, na foldu i

$p_i = p_i^1 - p_i^2$ - razlika na foldu i

$H_0: \mu_0 = 0$ vs $H_1: \mu_0 \neq 0$

– nul hipoteza je da je srednja vrijednost $p_i = 0$

$$m = \frac{\sum_{i=1}^K p_i}{K} \quad s^2 = \frac{\sum_{i=1}^K (p_i - m)^2}{K - 1}$$

$$\frac{\sqrt{K}(m - 0)}{s} = \frac{\sqrt{K}m}{s} \sim t_{K-1}$$

- Dvo-strani test: $H_0: \mu_0 = 0$

$$\text{ako } \frac{\sqrt{K}m}{s} \in (-t_{\frac{\alpha}{2}, 5}, t_{\frac{\alpha}{2}, 5})$$

Usporedba dva algoritma (3)

5x2struka unakrsna validacija - **Dietterichov test**

$$H_0: \mu_0 = \mu_1 \text{ vs } H_1: \mu_0 \neq \mu_1$$

$p_i^{(j)}$: razlika u broju grešaka algoritma 1 i 2, na foldu $(j), j=1,2$, u replici $i=1,5$

$$\bar{p}_i = \frac{p_i^{(1)} + p_i^{(2)}}{2}$$

$$s_i^2 = \left(p_i^{(1)} - \bar{p}_i\right)^2 + \left(p_i^{(2)} - \bar{p}_i\right)^2$$

$$\frac{p_1^{(1)}}{\sqrt{\sum_{i=1}^5 s_i^2 / 5}} \sim t_5$$

Dvo-strani test - za neki α , $H_0: \mu_0 = \mu_1$ ako $\frac{p_1^{(1)}}{\sqrt{\sum_{i=1}^5 s_i^2 / 5}} \in (-t_{\frac{\alpha}{2}, 5}, t_{\frac{\alpha}{2}, 5})$

Usporedba > 2 algoritma na više nezavisnih skupova podataka

primjene:

- usporedbe novog algoritma sa drugim pristupima
- novi problemi => selekcija najboljeg modela

problem:

L algoritama, K skupova podataka ($K \cdot \text{treniranje}$ i $K \cdot \text{testiranje}$) (L grupa sa K vrijednosti)

- Treba usporediti L uzoraka i ustanoviti da li su razlike između algoritama statistički značajne.

pristup: Analiza varijance (ANOVA)

Nul hipoteza:

$$H_0: \mu_0 = \mu_1 = \mu_2 = \dots = \mu_L \text{ vs } H_1: \mu_i \neq \mu_j - \text{za barem jedan par algoritama } (r,s)$$

Kritika „frequentist” pristupa usporedbi algoritama

- Ne znamo veličinu efekta (stvarnu razliku)
- statistička signifikantnost \neq praktička značajnost

➔ Bayesovski pristup – usporedbe i testovi na osnovu distribucija razlika



Mjere/statistike uspješnosti u klasifikaciji

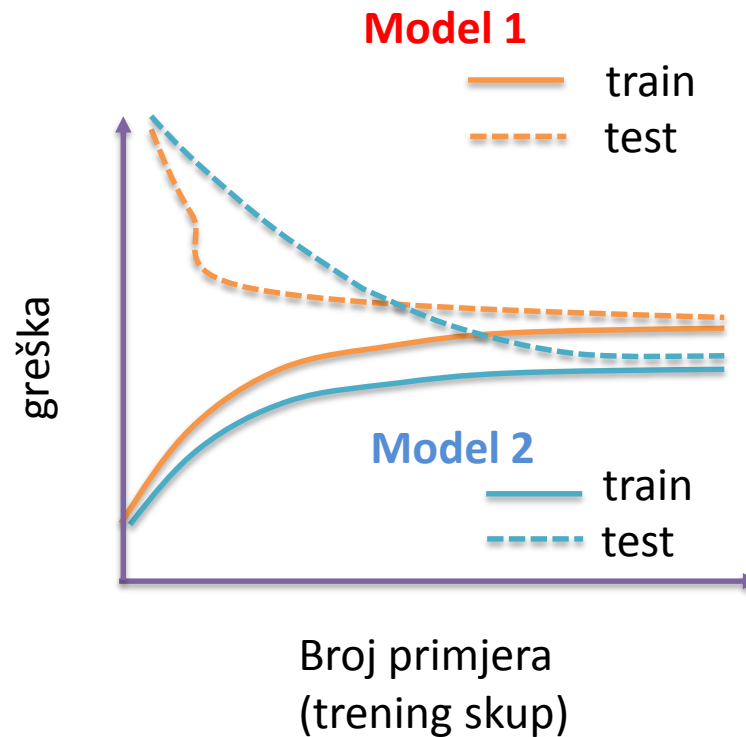
- ☐ Neke od mjera/statistika postoje u principu od vremena prije nastanka pojma strojno učenje
- ☐ Osim SU neke od ovih mjera su uobičajene i u području IR (en. information retrieval)
- ☐ Neke od mjera uspješnosti modela
 - Krivulja (točnosti) učenja
 - Matrica konfuzije
 - Točnost, Osjetljivost, Preciznost (Recall/Precision)
 - F_1 , ROC – Receiver Operating Curve - (AUC)
 - Kappa

Problem – Glasači vs. Ne-glasači (HR)

- Godine: ('18-25','26-35','36-45',...,76+)
 - Spol: {M, Ž}
 - Brak: {Da, Ne}
 - Obrazovanje: {nš,oš,sš,vš,vss}
 - Broj djece: ('0','<=2','3+')
 - Regija: {I,S,J,Z,C}
- Primanja {<50, 50-100, 100-200, >200}
 - Zaduženost (kredit): {0-50, 50-100, 100-200, >200}
 - Najčešće čita novine {V, JL, M, SN, Os}
 - Klasa: {1,0}

Godine	Spol	Brak	Obrazovanje	Broj djece	Regija	Primanja (kHRK/god)	Zaduženost (kHRK)	Novine	Klasa G(+)/ NG(-)
'26-35'	m	Da	sš	'<=2'	I	'50-100'	'50-100'	V	-
'26-35'	ž	Ne	vss	'0'	S	'<50'	'0-50'	JL	+
'56-65'	ž	Da	ss	'3+'	J	'100-200'	'100-200'	Os	-
'66-75'	m	Ne	vss	'<=2'	Z	'<50'	'>100'	M	+
'18-25'	m	Ne	ss	'0'	C	'<50'	'0-50'	SN	+
.....

Krivulja učenja



Naš naučeni model h aproksimira ciljnu funkciju f koja preslikava \mathbf{x} ($G, S, B, O, BrD, R, P, Z, N$) u Klasu $\{+, -\}$

Matrica konfuzije (confusion matrix)		Stvarna klasa	
		G (+) Pozitivni	NG (-) Negativni
Predviđeno modelom h	G (+) Pozitivni	TP	FP
	NG (-) Negativni	FN	TN

TP - true positives (broj stvarno pozitivnih primjera, točno predviđenih od strane modela h)

FP - false positives (broj stvarno negativnih primjera, koji su netočno predviđeni od strane modela h kao pozitivni)

TN – true negatives (broj stvarno negativnih primjera, koji su točno predviđeni od strane modela h kao negativni)

FN – false negatives (broj stvarno pozitivnih primjera, koji su netočno predviđeni od strane modela h kao negativni)

Matrica konfuzije (confusion matrix)		Stvarna klasa	
		G (+) Pozitivni	NG (-) Negativni
Predviđeno modelom <i>h</i>	G (+) Pozitivni	TP	FP
	NG (-) Negativni	FN	TN

Točnost (en. Accuracy) = $(TP+TN) / (TP+FP+TN+FN)$

Omjer točno klasificiranih primjera u odnosu na ukupan broj primjera (recimo u testnom skupu primjera)

❑ vrlo česta i uobičajena mjera - ne uvijek i ono što nam treba:

- pozitivni primjeri su nam daleko važniji (medicina)
(točnost daje *istu težinu* i pozitivnim i negativnim primjerima)
- u mnogim problemima imamo veliki nesrazmjer između broja pozitivnih i negativnih primjera

Matrica konfuzije (confusion matrix)		Stvarna klasa	
		G (+) Pozitivni	NG (-) Negativni
Predviđeno modelom <i>h</i>	G (+) Pozitivni	TP	FP
	NG (-) Negativni	FN	TN

Osjetljivost (en. Sensitivity / Recall / True positive rate)

$$R = TP / (TP + FN)$$

Udio točno pozitivnih primjera koje je model prepoznao kao pozitivne, od ukupnog broja pozitivnih primjera.

- npr. Moramo imati $R \approx 1$ – da ne bi “ispustili” teškog bolesnika

Specifičnost = $TN / (FP + TN)$ $S \approx 1 = P$ [Test je negativan | Pacijent je zdrav]

Preciznost

$$P = TP / (TP + FP)$$

Udio stvarno pozitivnih primjera u svima koji su modelom predviđeni kao pozitivni => pretraživači/preporučitelji - Information Retrieval (IR)

Evaluacija IR (information retrieval) sistema (npr. pretraživači)

- Učinkovitost (Effectiveness)
 - Pronalaženje **relevantnog** sadržaja (dokumenata iz korpusa – npr. Internet)
 - Ostali elementi
 - Efikasnost (indeksiranje, brzina)
 - Ekspresivnost (kompleksnost informacija)
 -

IR - Kvantifikacija relevantnosti

- ~ klasifikacija (važni/nevažni)
 - Pretraživač (~ klasifikator)
(problem: razlučiti važno/nevažnog)
 - korpus dokumenata = testni skup

IR - Kvantifikacija relevantnosti

- IR sistem vrati određeni skup dokumenata
- Možemo opet upotrijebiti matricu konfuzije !

Matrica konfuzije (confusion matrix)		Stvarna klasa	
		V (+) Važni	NV (-) NeVažni
Predviđeno pretraživačem	G (+) Važni	TP	FP
	NG (-) NeVažni	FN	TN

Točnost i IR

- Točnost: $Acc = (TP+TN) / (TP+FP+TN+FN)$

= Udio točnih klasifikacija

– Za IR – neupotrebljivo !

|Važno| <<< |Nevažno|

TP <<< TN

$$Acc = (1 + 997) / (1+1+997+1) = 99.8\% \Rightarrow$$

Matrica konfuzije (confusion matrix)		Stvarna klasa	
		V (+) Važni	NV (-) NeVažni
Pretraživač	G (+) Važni	1	1
	NG (-) NeVažni	1	997

$$Acc = (0 + 998) / (0+0+998+2) = 99.8\% \Rightarrow$$

Matrica konfuzije (confusion matrix)		Stvarna klasa	
		V (+) Važni	NV (-) NeVažni
Pretraživač	G (+) Važni	0	1
	NG (-) NeVažni	2	997

- Preciznost $P = tp/(tp+fp)$
 - Udio važnih dokumenata od onih koji su “pronađeni” (klasificirano kao važni !)
 - P [*pronađeni važni* | *ukupno “pronađeni”*]
- Osjetljivost – recall $R = tp/(tp+fn)$
 - Udio od ukupno važnih koji su i pronađeni kao važni
 - P [*pronađeni važni* / *ukupno važnih*]
 - R može biti jednak 1 - ali je tada preciznost loša !

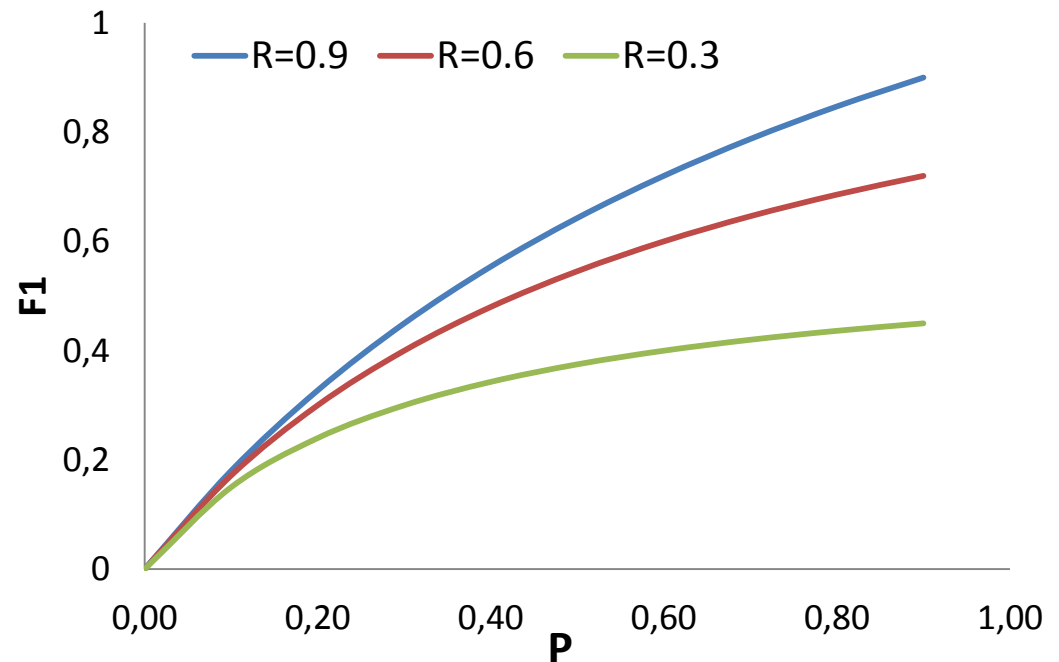
- Dobar IR
 - Balans Preciznost/Osjetljivost
 - Tipično:
 - preciznost pada kako osjetljivost raste, i obratno
 - F_β : tzv. F-mjera koja povezuje Preciznost/Osjetljivost
 F_β : harmonijska sredina P i R - uz težinski faktor β

$$F_\beta = \frac{1}{\alpha \frac{1}{P} + (1-\alpha) \frac{1}{R}} = \frac{(\beta^2 + 1)PR}{\beta^2 P + R}$$

Uobičajeno se u IR koristi :

F_1 : Balansirani P i R (tj $\beta = 1$ ili $\alpha = \frac{1}{2}$)

$$F_1 = \frac{2}{\frac{1}{P} + \frac{1}{R}} = \frac{2PR}{P + R}$$



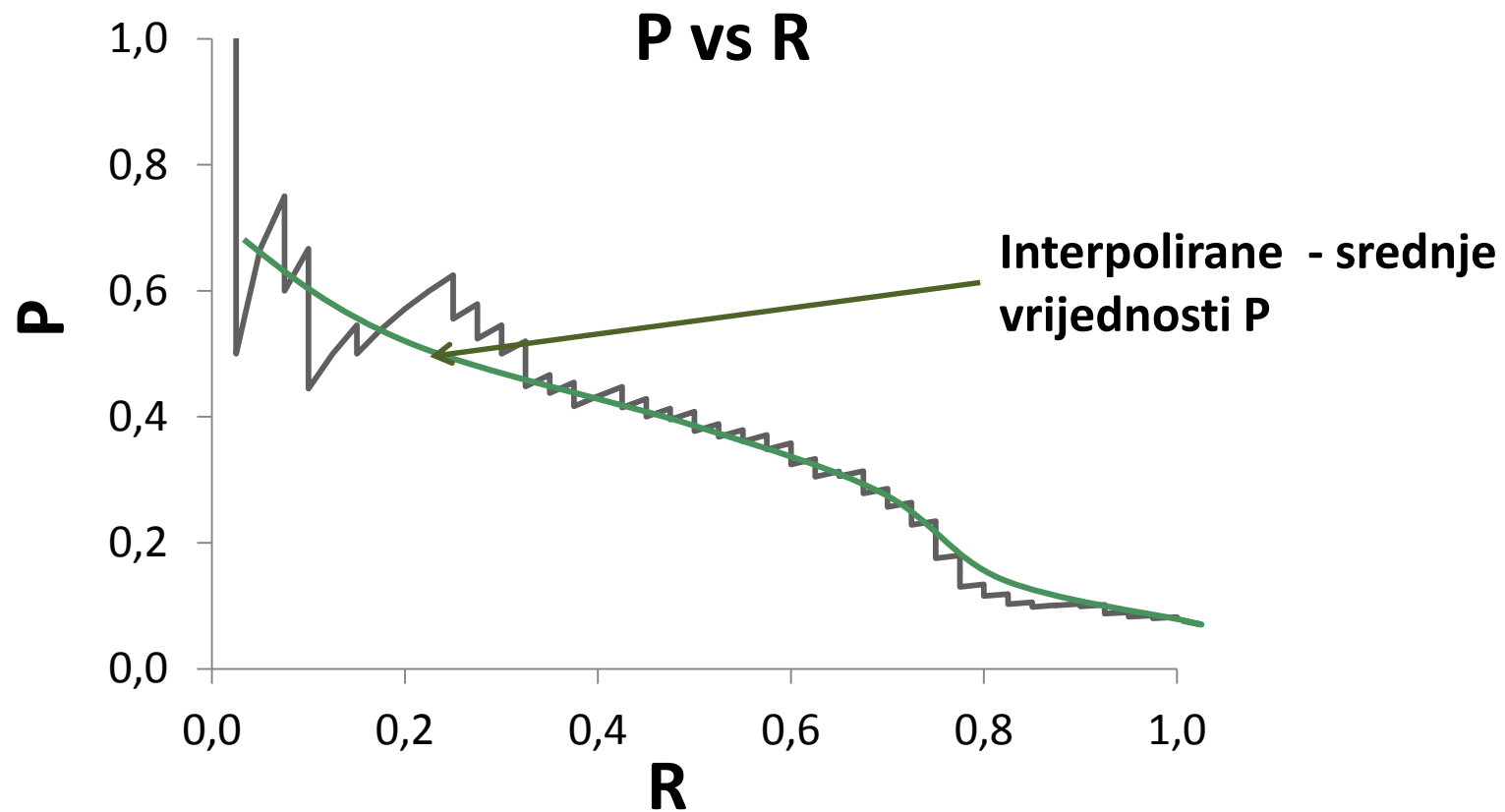
Ako **IR-sistem** rangira dokumente:

R, P, F_1 - f (odabranog broja rangiranih dokumenata)

Ako **klasifikator** rangira primjere

npr. prema vjerojatnosti pripadanja određenoj klasi:

R, P, F_1 - f (odabranog broja rangiranih primjera)



Mjere vezane uz rangiranje primjera

- $P(k)$ – preciznost za top k rangiranih primjera
- $R(k)$ – osjetljivost za top k rangiranih primjera

ROC – krivulja

– prikazuje odnos TPR u odnosu na FPR

- *TPR* – broj korektnih klasifikacija u (pozitivnoj) klasi u odnosu na ukupan broj pozitivnih primjera

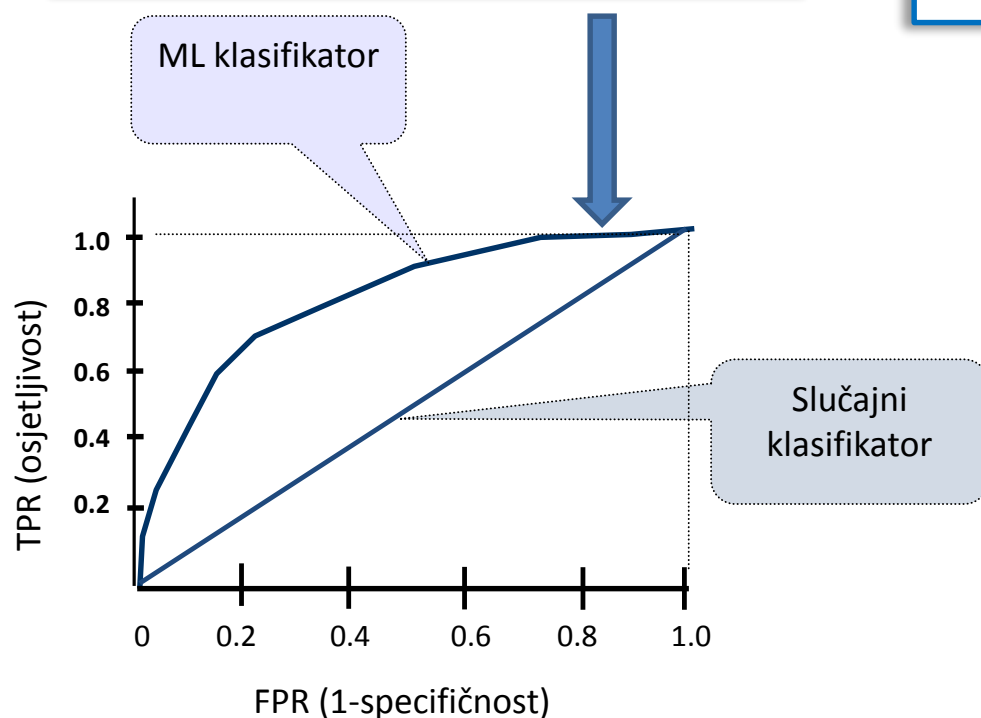
$$TPR = tp/(tp+fn) = R$$

- *FPR* – broj krivih klasifikacija u (pozitivnoj) klasi u odnosu na ukupan broj negativnih primjera

$$FPR = fp/(fp+tn) = 1 - \text{specifičnost}$$

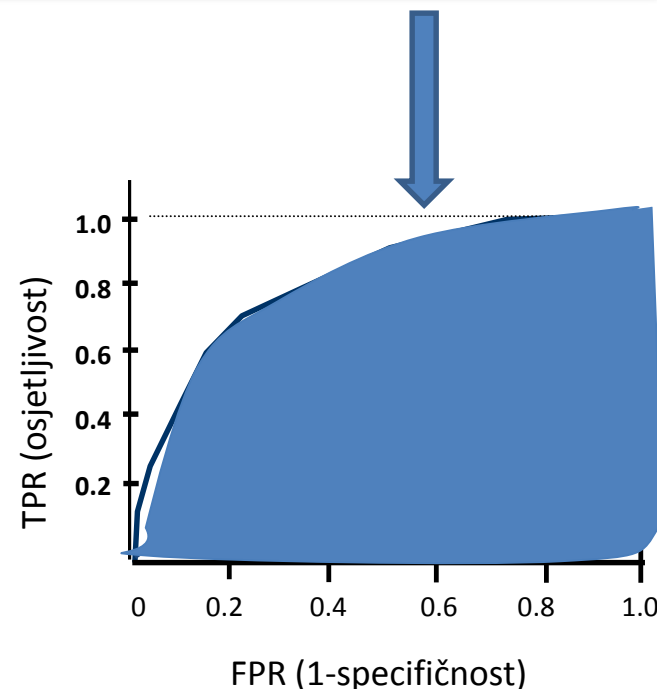
ROC – krivulja

Mjera uspješnosti klasifikatora
– na nivou jedne klase primjera !



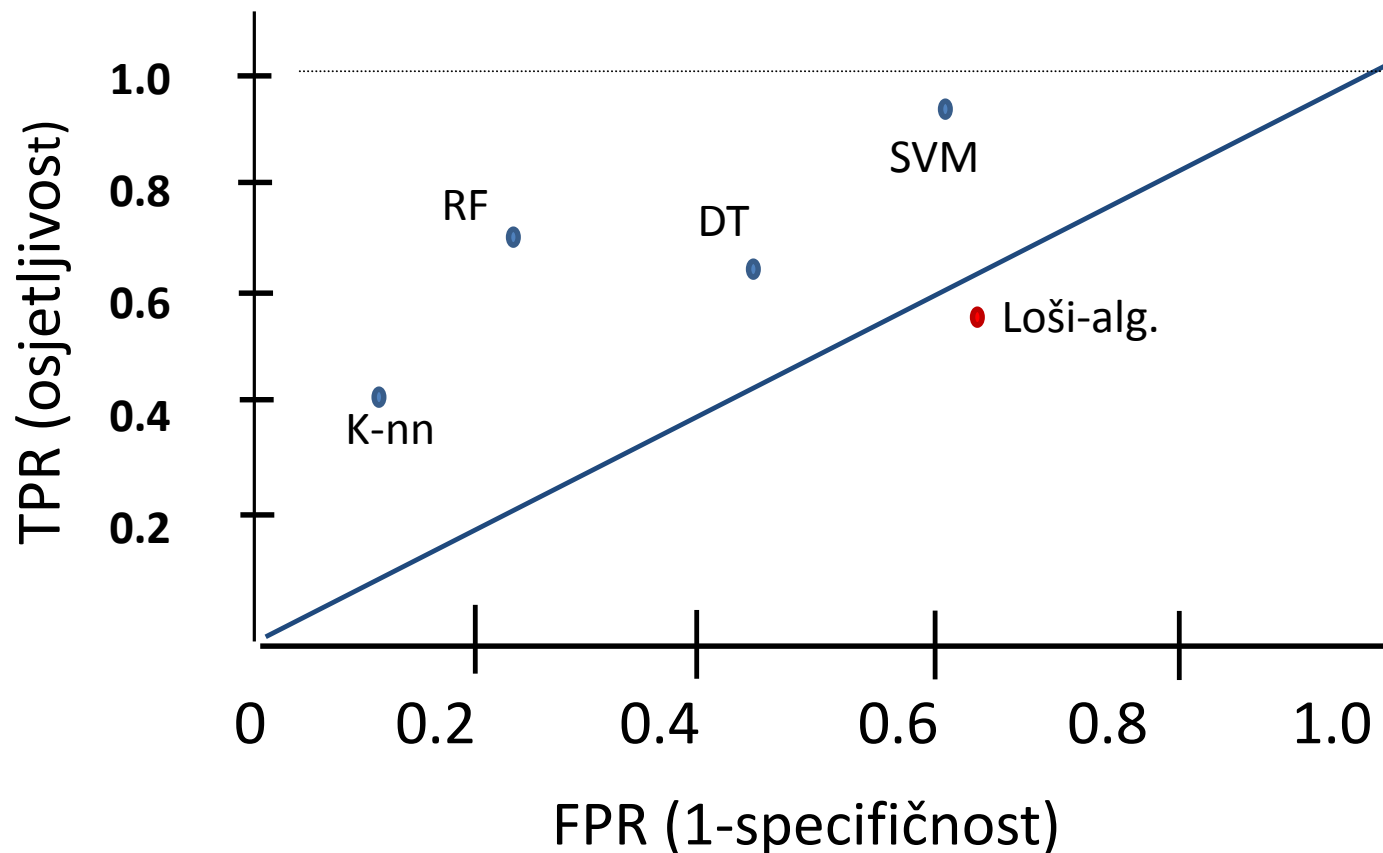
AUC - Površina ispod ROC krivulje (en. AUC – Area Under Curve)

- AUC=0.5 – slučajno „pogađalo”
- AUC=1 – savršeni klasifikator



ROC -prostor za komparaciju hipoteza/algoritama

Nemamo rangirane primjere - h u točki (TPR,FPR)



Kappa - mjera slaganja eksperata u predikciji (Inter-Judge Agreement)

$$\kappa = \frac{P(A) - P(S)}{1 - P(S)}$$

$P(A)$: proporcija primjera kod kojih se eksperti slažu

$P(S)$: očekivana proporcija primjera kod kojih se slaganje postiže slučajnim predikcijama

- $\kappa = 0$: sasvim slučajno slaganje;
- $\kappa = 1$: savršeno slaganje;
- $\kappa > 0.8$: dobro slaganje
- $0.67 < \kappa < 0.8$: tentativno slaganje

Kako to funkcionira – slaganje eksperata ?

Broj primjera	Ekspert 1	Ekspert 2
200	važno	važno
50	nevažno	nevažno
30	važno	nevažno
20	nevažno	važno

$$P(E1=E2) = (200+50)/300 = 0.83$$

$$P(\text{nevažno}) = (20+30+50+50)/(300+300) = 0.25$$

$$P(\text{važno}) = (20+30+200+200)/(300+300) = 0.75$$

$$P(\text{slučajno}) = 0.25^2 + 0.75^2 = 0.63$$

$$\kappa = (0.83 - 0.63)/(1-0.63) = 0.54$$

Kako to funkcionira - u klasifikaciji?

		Stvarna klasifikacija - ekspert 1	
Predikcija Klasifikatora - ekspert 2		v	n
	v	200	20
	n	30	50

$$P(E1=E2) = (200+50)/300 = 0.83$$

$$P(\text{nevažno}) = (20+50)/(300) = 0.25$$

$$P(\text{važno}) = (30+200)/(300) = 0.75$$

$$P(\text{slučajno}) = 0.25^2 + 0.75^2 = 0.642$$

Što pretpostavlja
 $P(\text{slučajno})$?

$$\kappa = (0.83 - 0.63)/(1-0.63) = 0.53$$

Sažetak

Teorijske procjene greške

Resampling metode

Statistička evaluacija

odabir/usporedba modela i algoritama

Mjere uspješnosti u klasifikaciji

Literatura:

Evaluacija modela:

Machine learning

T. Mitchel (ch. 5)

The Elements of Statistical Learning

Hastie, Tibshirani, Friedman (ch. 7)

ROC (Receiver Operating Characteristic)

P.A. Flach – ROC tutorijali + članci

Model selection (statistička evaluacija, usporedba modela/algoritama):

T Dietterich, Approximate statistical tests for comparing supervised classification learning algorithms, Neural Computation, (1998), 10, 1895-1923
Introduction to Machine Learning - E. Alpaydin (ch. 19)

A Benavoli et al, Time for a Change: a Tutorial for Comparing Multiple Classifiers Through Bayesian Analysis, JMLR, (2017)

Priistranost modela i varijanca (bias & variance)

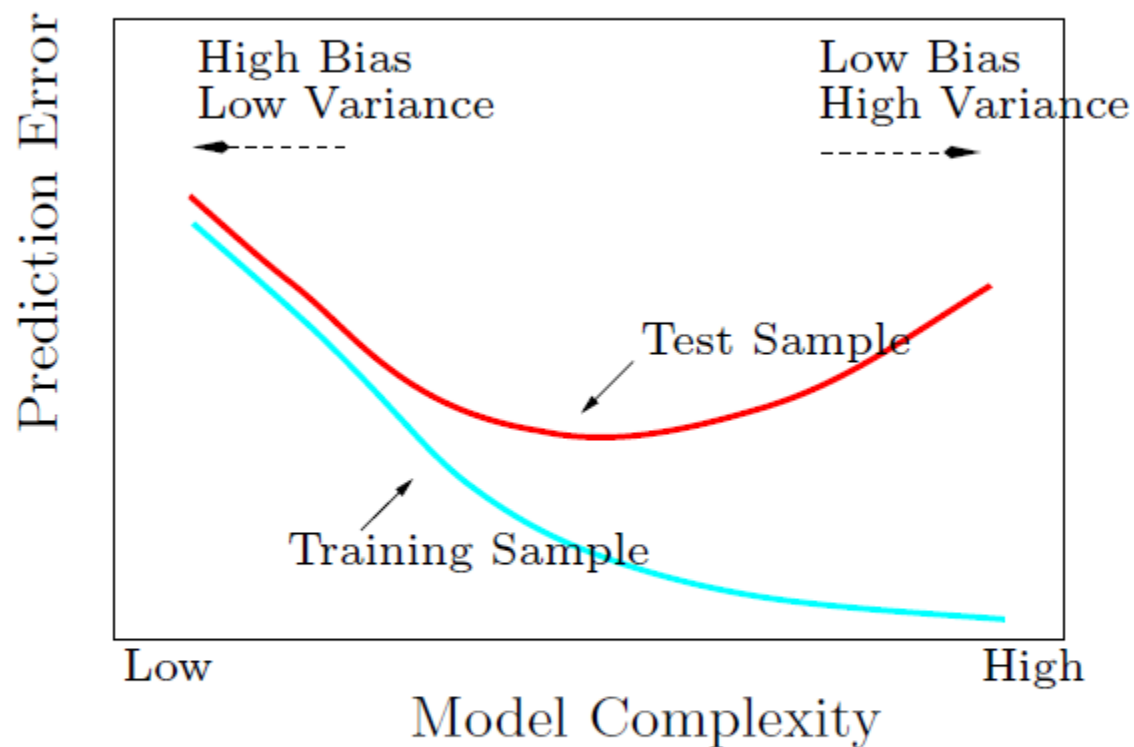


FIGURE 2.11. *Test and training error as a function of model complexity.*

Kod određivanja aproksimacije funkcije

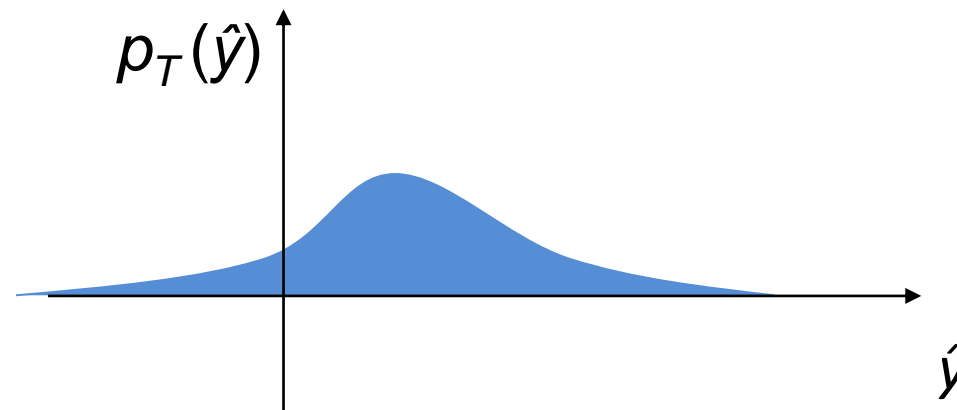
– ciljna varijabla y se obično može izraziti kao:

$$y(\mathbf{x}) = f(\mathbf{x}) + \varepsilon$$

- $f(\mathbf{x})$ - “ciljna funkcija”
- ε - šum; obično $E(\varepsilon / \mathbf{x}) = 0$

Uz određene modifikacije slijedeće razmatranje se može promijeniti i na klasifikacijske probleme

Skup za učenje je T slučajno uzorkovan
=> predikcija \hat{y} slučajna varijabla



Dekompozicija prediktivne pogreške

Pristranost i varijanca

$$Y = f(X) + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \sigma_\varepsilon)$$

$$SE(x) = E[(Y - \hat{f}(x))^2] \quad \text{Očekivana vrijednost kvadratne pogreške}$$

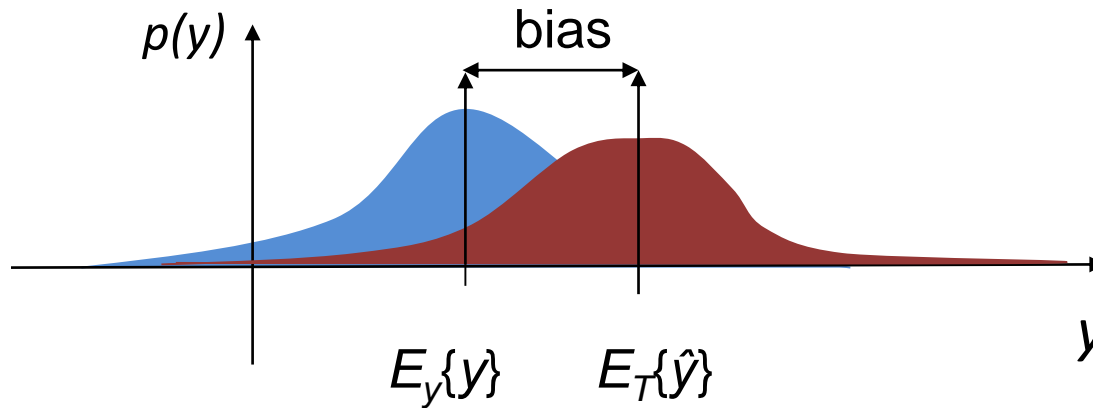
$$SE(x) = \underbrace{(E[\hat{f}(x)] - f(x))^2}_{\text{Pristranost}} + \underbrace{E[(\hat{f}(x) - E[\hat{f}(x)])^2]}_{\text{varijanca}} + \underbrace{\sigma_\varepsilon^2}_{\text{ne-reducibilna greška}}$$

Očekivane vrijednosti

$$E(e) = \text{pristranost} + \text{varijanca} + \text{šum}$$

- | | a | b | c |
|---------------------|---|---|---|
| • Pristranost/Bias: | sistematska greška na točki x - prosjek preko “svih” skupova za učenje T veličine N | | |
| • Varijanca: | Varijacija greške oko prosječne vrijednosti | | |
| • Šum: | Greška u određivanju stvarnih vrijednosti $f(x)$ | | |

Dekompozicija prediktivne pogreške: Pristranost i varijanca modela

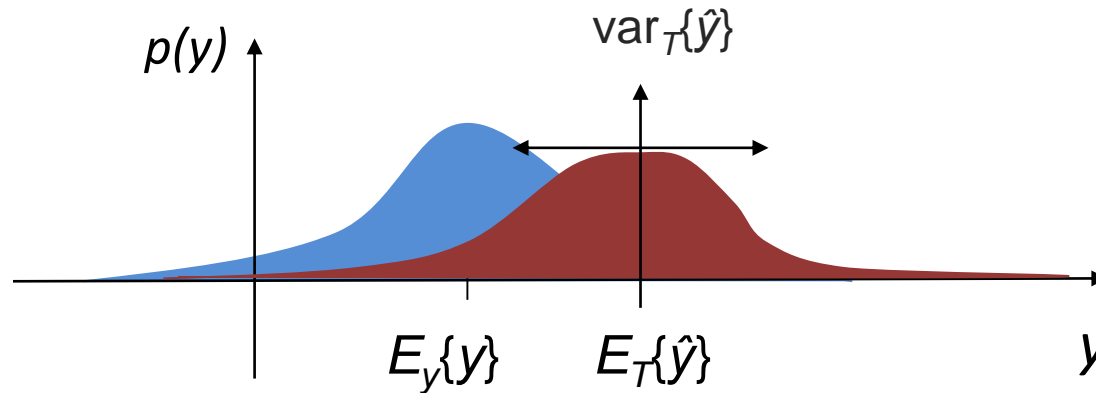


$$(E_y\{y\} - E_T\{\hat{y}\})^2$$

$E_T\{\hat{y}\}$ = prosječni rezultat modela (preko svih T)

bias = greška između stvarne vrijednosti i prosječnog estimacijskog modela

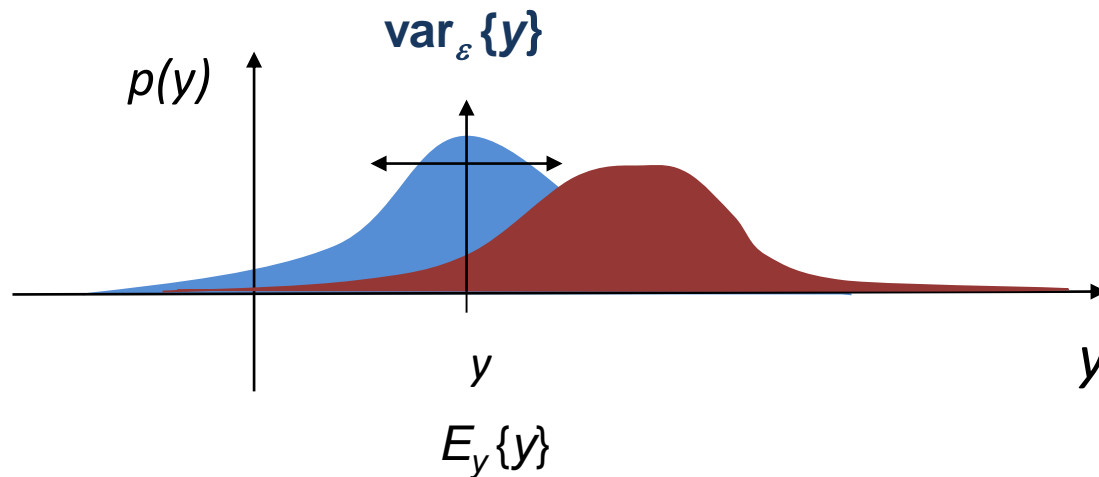
Dekompozicija prediktivne pogreške: Pristranost i varijanca modela



$$\text{var}_T\{y\} = E_T\{(\hat{y} - E_T\{\hat{y}\})^2\}$$

$\text{var}_T\{\hat{y}\}$ = estimacijska varijanca = zbog over-fitinga

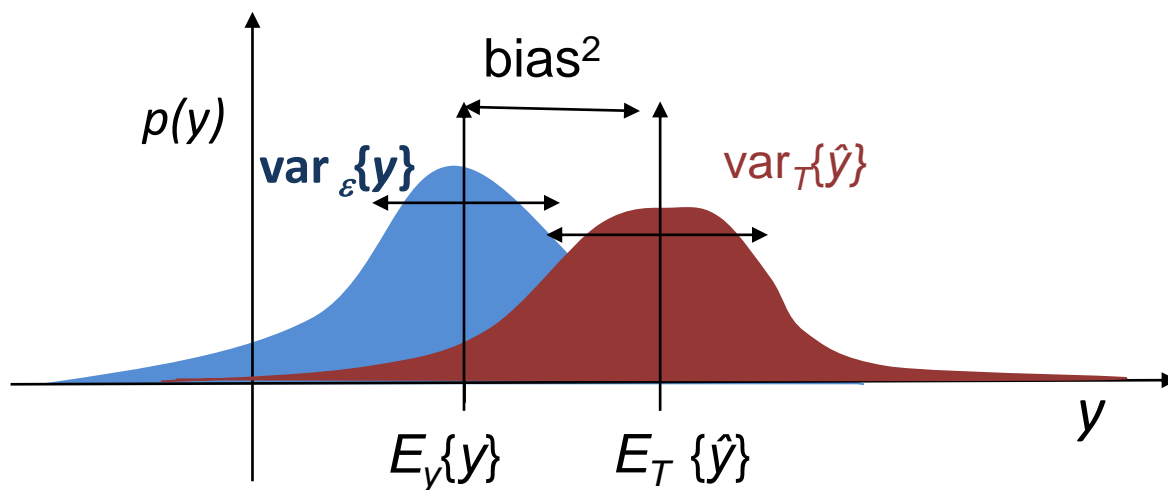
Dekompozicija prediktivne pogreške: Pristranost i varijanca modela



$$\text{var}_\epsilon\{y\} = E_y\{(y - E_y\{y\})^2\}$$

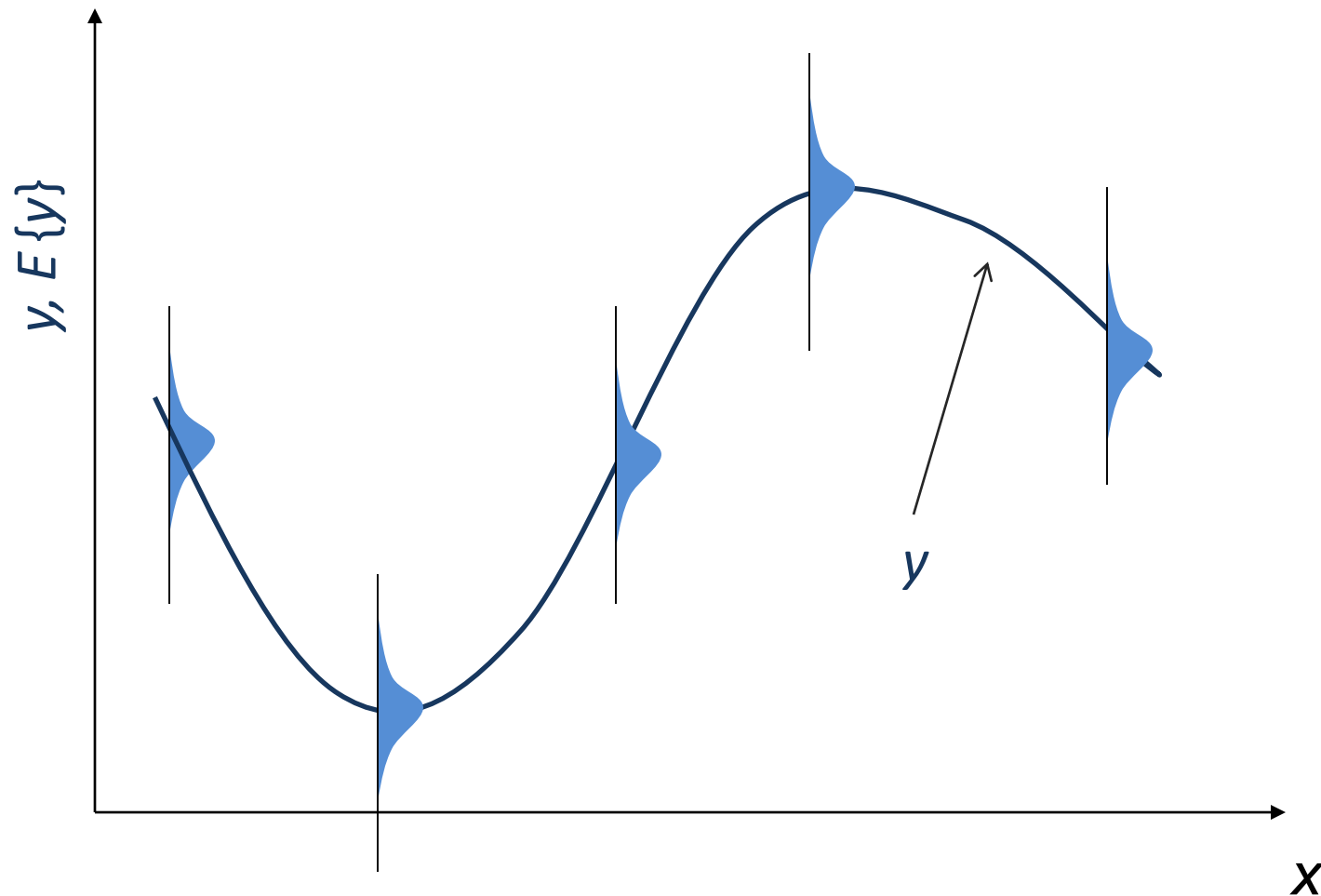
rezidualna greška = minimalna greška koju možemo dostići

Dekompozicija prediktivne pogreške: Pristranost i varijanca modela

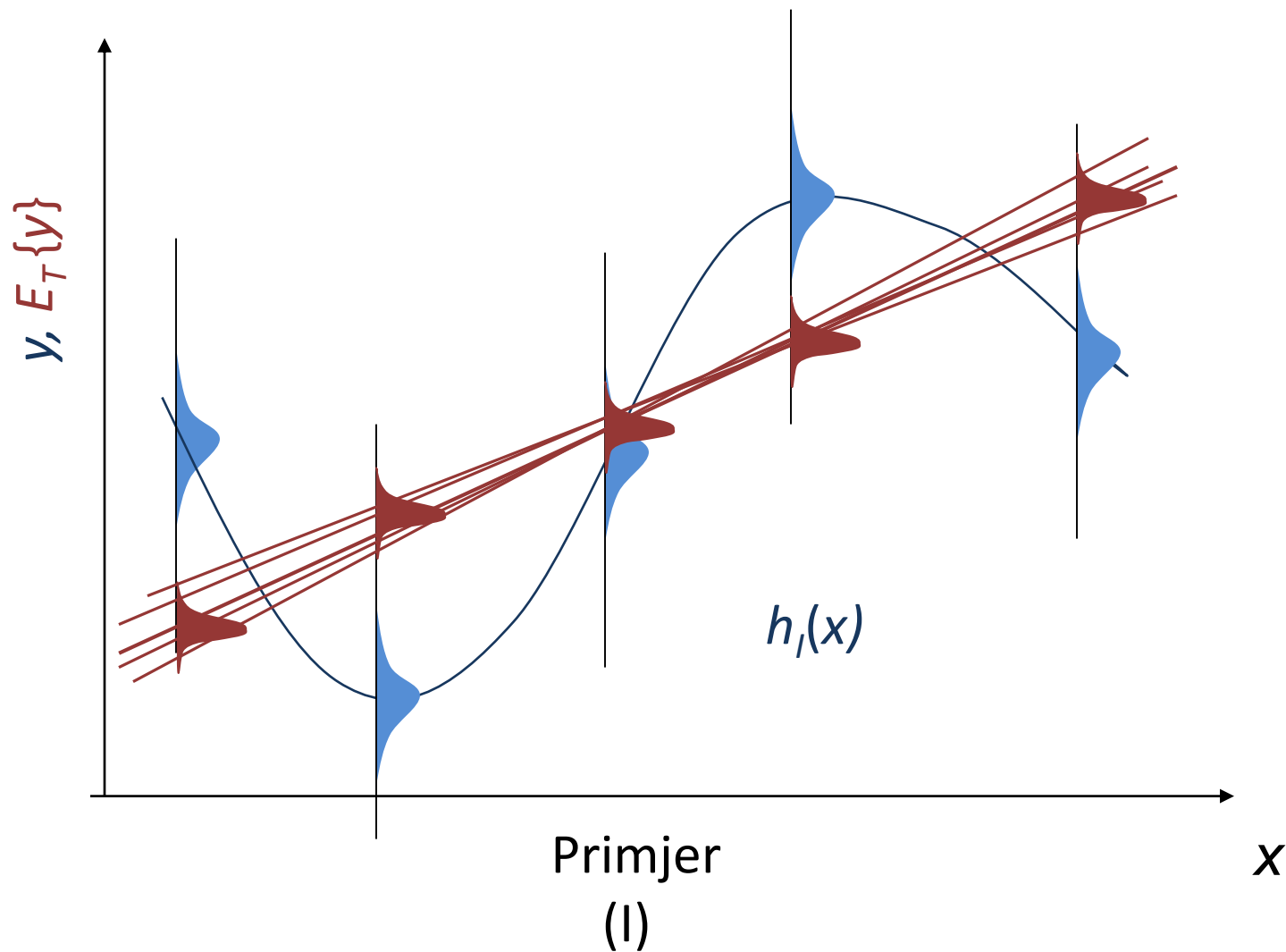


$$E = \text{var}_\varepsilon\{y\} + \text{bias}^2 + \text{var}_T\{\hat{y}\}$$

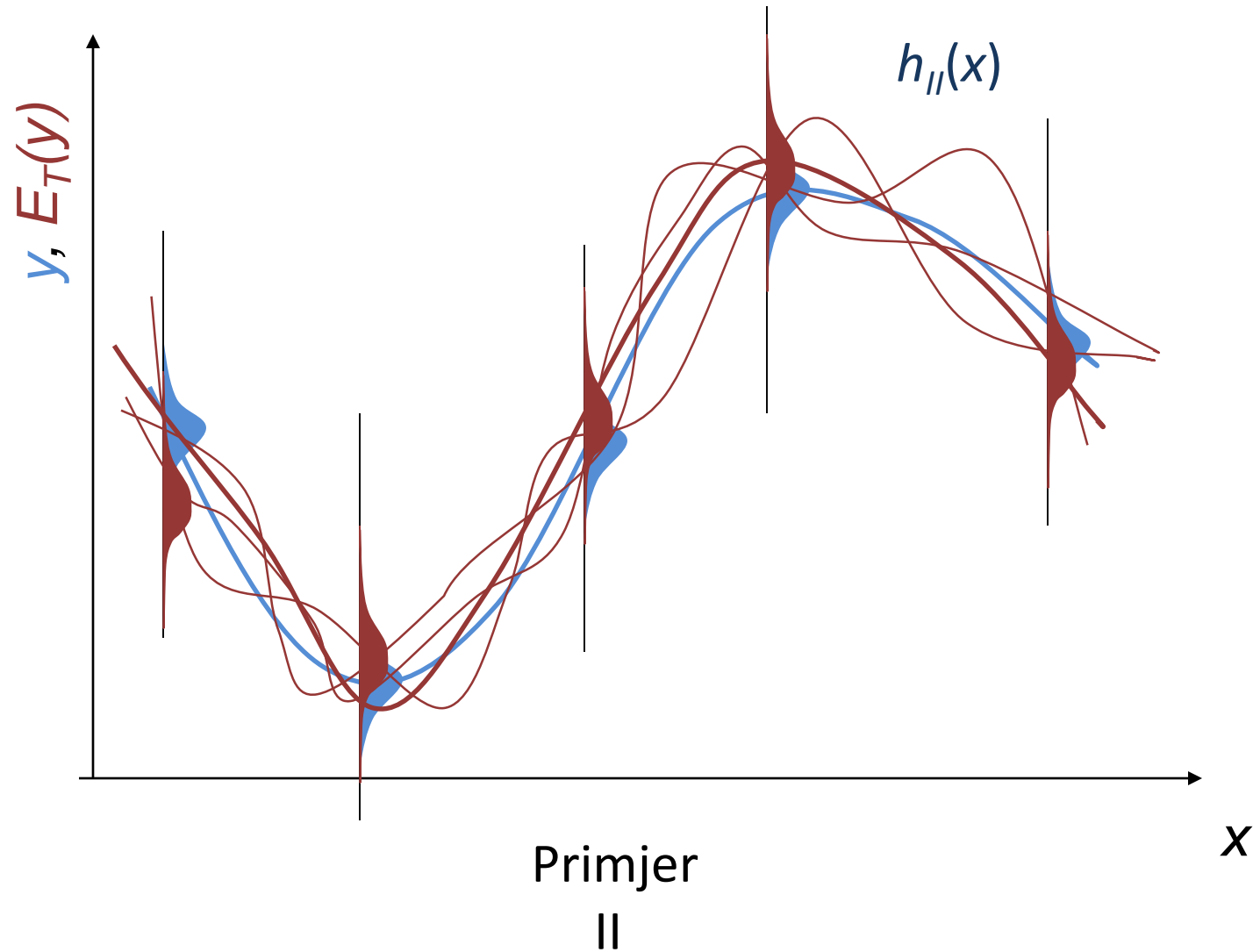
Dekompozicija prediktivne pogreške: Pristranost i varijanca modela



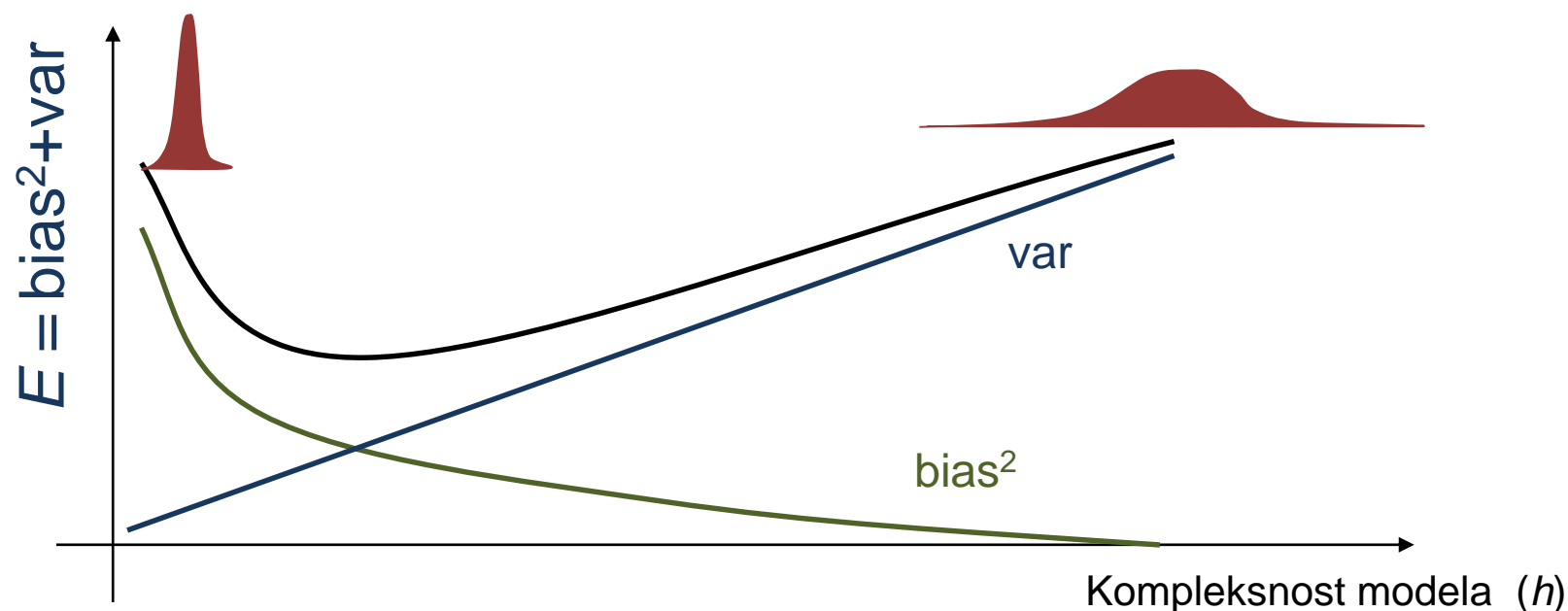
Dekompozicija prediktivne pogreške: Pristranost i varijanca modela



Dekompozicija prediktivne pogreške: Pristranost i varijanca modela



Dekompozicija prediktivne pogreške: Pristranost i varijanca modela

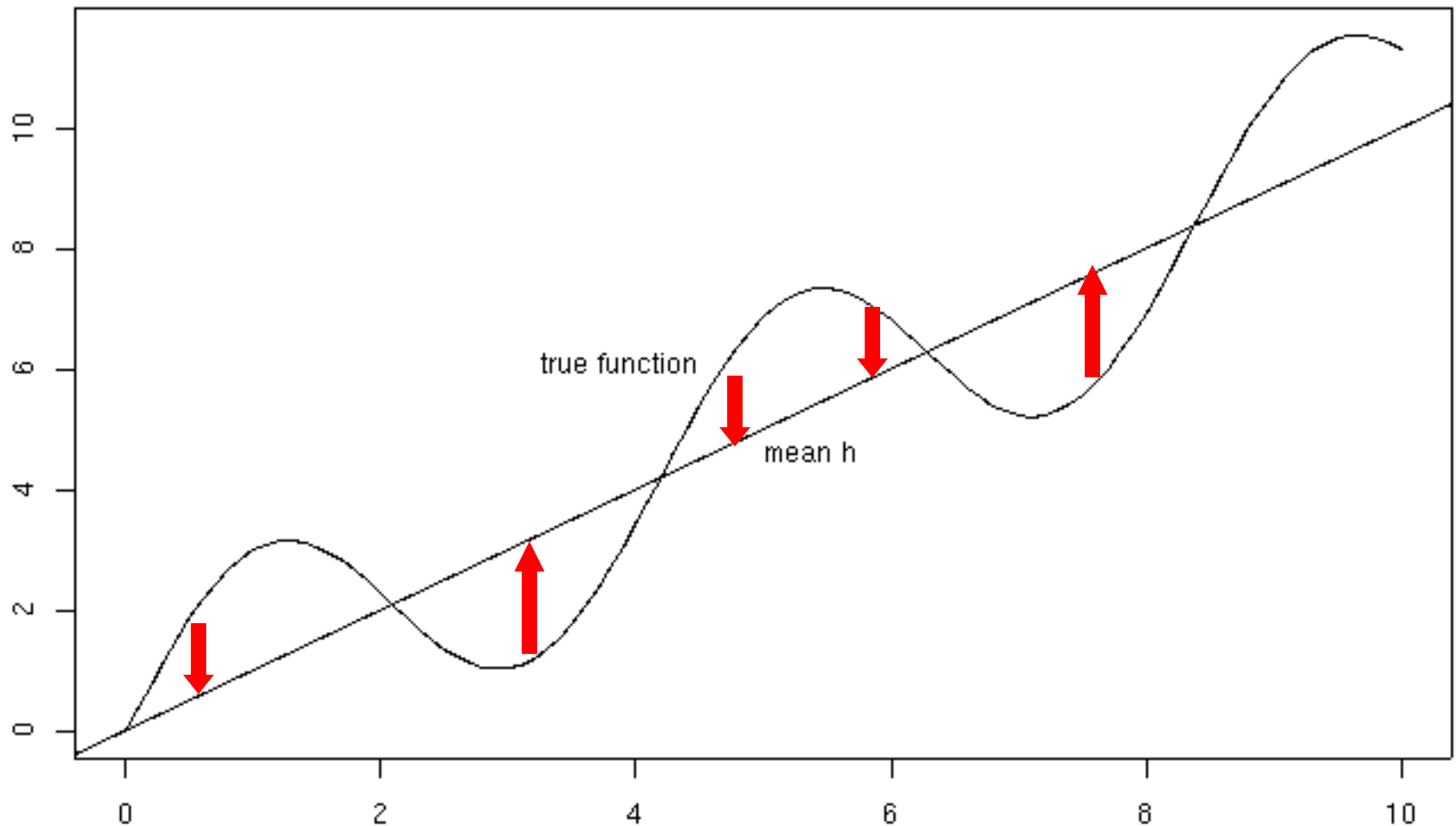


Pristranost (bias) obično pada s povećanjem kompleksnosti modela, dok se varijanca povećava s kompleksnosti modela

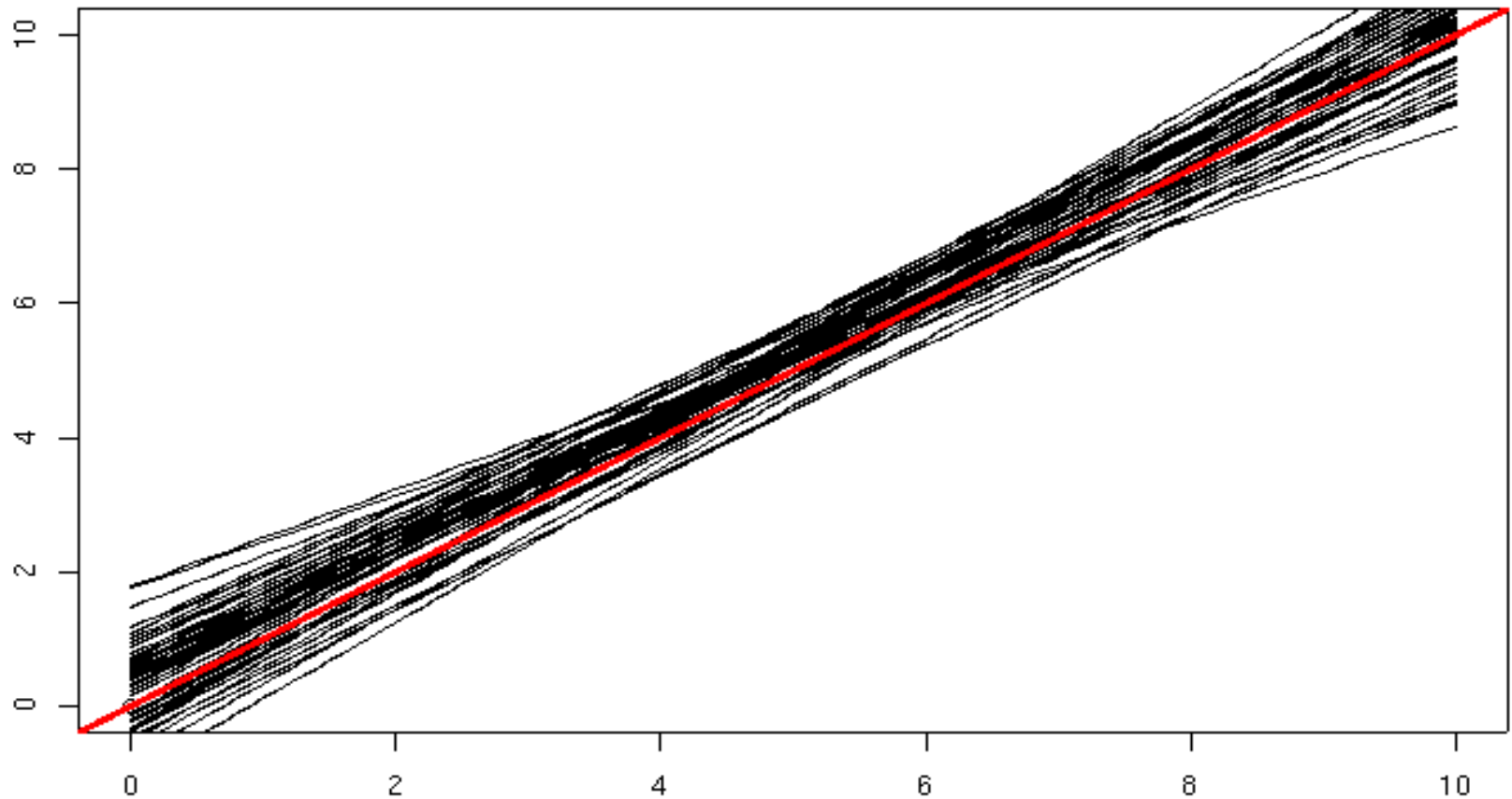
Mjerenje pristranosti (bias) i varijance

- Pristranosti i varijanca – definirani su kao očekivanja !
- Da bi se odredili moramo simulirati velik broj skupova T
- Na taj način možemo odrediti i velik broj modela – te ih iskoristiti za određivanje prosječnih vrijednosti

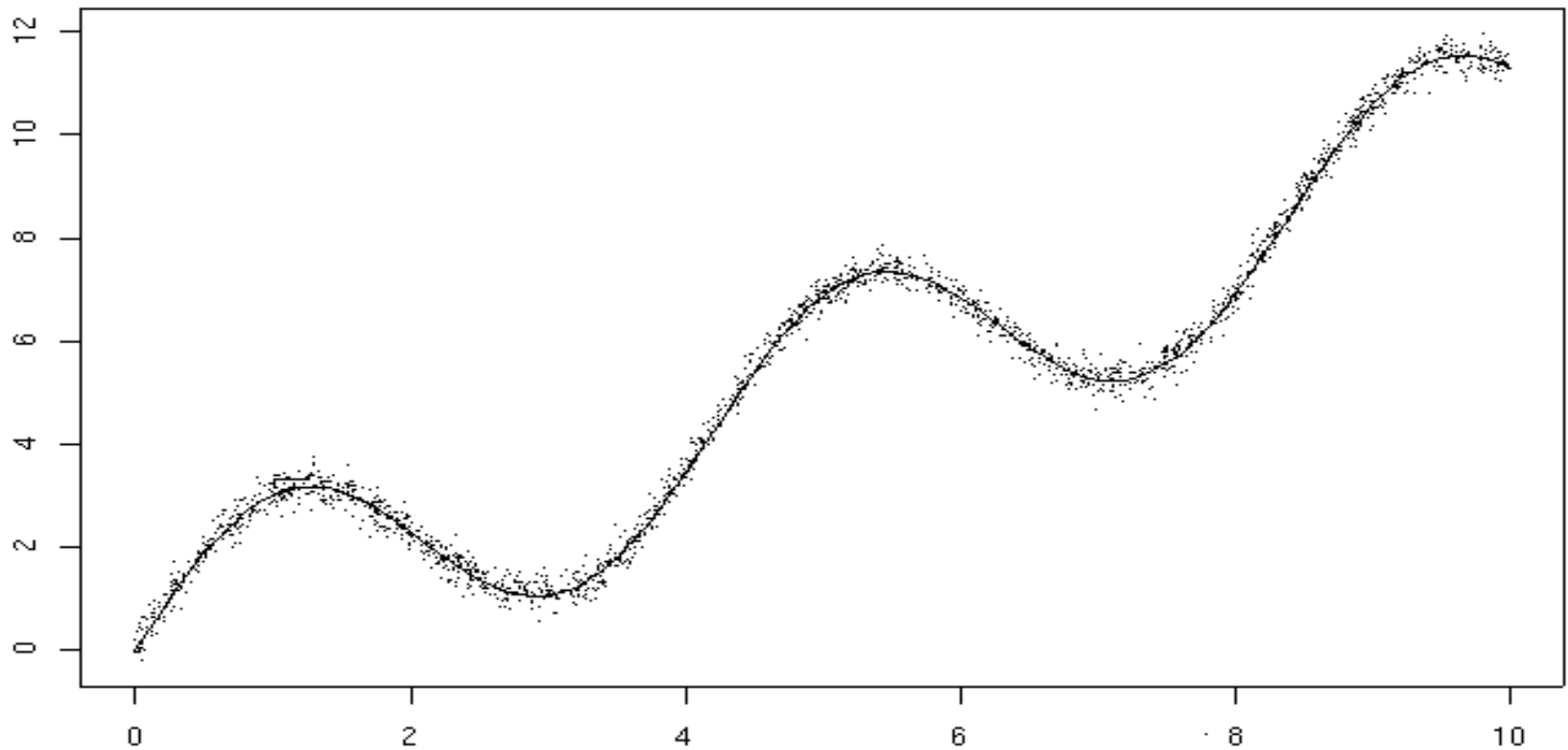
Pristranost modela (Bias)



Varijanca



Šum



Bootstrap pristup

- 1 Uz dani skup podataka D , odvojimo (1/3) podataka u skup za testiranje (D_h – hold-out set), a preostale ostavimo u D_t ;
- 2 Iz skupa D_t (veličine N), konstruiramo tzv. “bootstrap” repliku skupa - D_b , tako da slučajno uzorkujemo N primjera iz D_t (uz dozvoljeno višestruko izvlačenje istog primjera!) ;
- 3 Algoritmom strojnog učenja konstruiramo model h_b , treniranjem na D_b
- 4 Korištenjem h_b odredimo predikcije na primjerima iz D_h , te odredimo grešku
- 5 Ovaj proces se tipično ponovi velik broj puta ($K > 30$)

Određivanje pristranosti(bias) i varijance korištenjem bootstrap uzoraka

1 Za svaki \mathbf{x} – skup predikcija $h_1(\mathbf{x}), h_2(\mathbf{x}), \dots, h_K(\mathbf{x})$

Prosječna predikcija:

$$\bar{h}(\mathbf{x}) = \frac{1}{K} \sum_{k=1}^K h_k(\mathbf{x})$$

2 Bias :

$$Bias(\mathbf{x}) = y - \bar{h}(\mathbf{x})$$

3 Varijanca:

$$Varijanca(\mathbf{x}) = \frac{1}{K-1} \sum_{k=1}^K (h_k(\mathbf{x}) - \bar{h}(\mathbf{x}))^2$$

Literatura:

- The Elements of Statistical Learning
Hastie, Tibshirani, Friedman (ch. 15)
- Introduction to Machine Learning
E. Alpaydin (ch. 19)
- AI – Modern approach
Russel & Norvig (ch 18.4)