Presentazione modulo *Machine Learning*

Luca Mastrobattista
0292461

`SonarCloud`:
https://sonarcloud.io/summary/overall?id=lucaMastro_deliverable2

`github`: https://github.com/lucaMastro/ISW2-deliverables

Contesto

Il *machine learnig* applicato all'ingegneria del software può avere diversi utilizzi:

- *Learning association*
- *Classification*
- ***Prediction***

In ogni caso, una cosa di cui non si può fare a meno è la raccolta di informazioni.

Scopo del progetto

Il progetto ha come obiettivo quello di confrontare l'accuratezza di 3 modelli predittivi differenti, utilizzando la stessa tecnica di valutazione *walk forward*, in funzione di tecniche diverse di:

- *sampling*
- *cost sensitive*
- *feature selection*

Le fonti per la raccolta dei dati saranno *Git* e *Jira*.

Progetti

I dati da utilizzare per i modelli dovranno essere recuperati dai progetti `BookKeeper` e uno ottenuto in base alla prima lettera del cognome, che nel mio caso risulta in `OpenJPA` (M $\equiv$ 13 $\rightarrow$ `X = 13 % 6 = 1`):

```
switch (X) {
  case 0: project == AVRO;
  case 1: project == OPENJPA;
  case 2: project == STORM;
  case 3: project == ZOOKEEPER;
  case 4: project == SYNCOPE;
  case 5: project == TAJO;
}
```

Tipo di bug

Poiché lo scopo del progetto è quello di fare previsioni su quali classi risultano essere *buggy* nel presente o nel futuro a partire dai dati del passato, è necessario recuperare le informazioni giuste da Jira:

```
Type == ‘‘defect’’ AND
(status == ‘‘Closed’’ OR status == ‘‘Resolved’’)
AND Resolution == ‘‘Fixed’’
```

Dove queste risultino insufficienti per la classificazione, il progetto prevede l'utilizzo dell'algoritmo Proportion per completare il labeling delle classi.

Features di valutazione

Le metriche utilizzate nel progetto sono le seguenti:

Metrica	Descrizione
`LOC`	Linee di codice presenti
`NR`	Numero di revisioni
`NFix`	Numero di bugs risolti
`NAuth`	Numero di autori che hanno lavorato alla classe
`LOC_added`	Somma su tutte le revisioni delle righe di codice aggiunte
`MAX_LOC_added`	La massima quantità di linee di codice aggiunte in una singola revisione
`Churn`	Somma delle differenze tra le linee aggiunte e quelle rimosse
`MAX_churn`	Il massimo valore di Churn fra tutte le revisioni
`Age`	Età del file in settimane (dall'inizio del progetto)

Creazione del dataset e funzionalità

Pro:

- Utilizzo di libreria `JGit` per analisi dei dati del version control system
- Rimozione linee di commento nel calcolo della metrica LOC
- Rimozione righe duplicate all'interno dei file `.arff`
- Rimozione dei commit di *revert*
- I progetti target non sono stati *hardcoded*: il codice è riutilizzabile per qualsiasi nuovo progetto
- Gestione dei file rinominati nel corso delle versioni
- Semplice interfaccia grafica elementare
- Possibilità di clonare un nuovo repository
- facilmente mantenibile grazie a una struttura modulare: è facile aggiungere o rimuovere classificatori e algoritmi

Contro:

- Creazione del dataset e elaborazione dei dati tramite *API* di Weka potrebbe richiedere molto tempo

Problemi

Il codice funziona perfettamente sull'ultima versione del progetto openJPA, ma non sull'ultima del progetto BookKeeper: viene sollevata un'eccezione in corrispendenza della combinazione delle seguenti tecniche, con training set composto da 1 o 2 versioni:

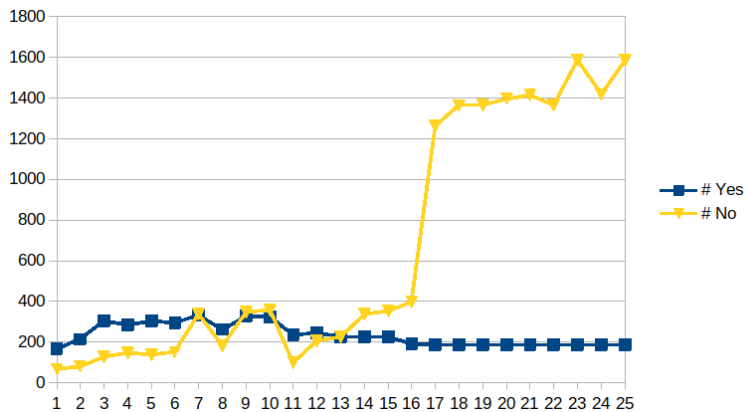
- *Sampling*: SMOTE
- *Features selection*: BEST FIRST
- *Cost sensitive classifier*: SENSITIVE LEARNING

I risultati presentati per il progetto BookKeeper non tratteranno questa combinazione.

Proportion classification

Come potevamo aspettarci, la percentuale dei difetti identificati diminuisce al crescere delle versioni, fino ad arrivare a una percentuale del solo 11% nell'ultima versione:

Version	# Yes	# No	% Yes	% No	nfix tot	# files
1	168	67	0,71	0,29	192	235
2	214	81	0,73	0,27	389	295
3	304	128	0,7	0,3	560	432
4	284	149	0,66	0,34	61	433
5	303	136	0,69	0,31	264	439
6	295	150	0,66	0,34	343	445
7	335	339	0,5	0,5	535	674
8	262	183	0,59	0,41	36	445
9	328	350	0,48	0,52	17	678
10	324	357	0,48	0,52	203	681
11	237	98	0,71	0,29	172	335
12	247	205	0,55	0,45	6354	452
13	226	227	0,5	0,5	536	453
14	225	340	0,4	0,6	0	565
15	226	354	0,39	0,61	0	580
16	192	398	0,33	0,67	6	590
17	188	1265	0,13	0,87	0	1453
18	188	1364	0,12	0,88	0	1552
19	188	1366	0,12	0,88	0	1554
20	188	1398	0,12	0,88	0	1586
21	188	1415	0,12	0,88	0	1603
22	188	1368	0,12	0,88	0	1556
23	188	1586	0,11	0,89	0	1774
24	187	1418	0,12	0,88	2	1605
25	187	1588	0,11	0,89	0	1775



Il numero di fix rimane a 0 da versione 14 in poi: questo spiega perché il numero di no aumenta drasticamente a release 17, dove vengono aggiunti anche molti files.

Massimi valori per i classificatori

L'obiettivo dell'analisi è valutare quali classificatori si comportano meglio in quali situazioni. Si ricerca per ogni metrica qual è il valore massimo assunto e quante volte questo si ripete.

Classificatore	Max Precision	Occorrenze Max Precision
RandomForest	1	5
NaiveBayes	1	27
Ibk	1	7

Classificatore	Max Recall	Occorrenze Max Recall
RandomForest	1	93
NaiveBayes	1	166
Ibk	1	36

Classificatore	Max AUC	Occorrenze Max AUC
RandomForest	0,978	1
NaiveBayes	0,985	1
Ibk	0,958	3

Classificatore	Max Kappa	Occorrenze Max Kappa
RandomForest	0,747	1
NaiveBayes	0,632	1
Ibk	0,783	5

Poiché analizzare tutte le combinazioni dei vari parametri sarebbe un lavoro lunghissimo, l'analisi si concentrerà fissando la variabile *dataset*; si analizzeranno i seguenti casi per il *training set*:

- 1 versione (step iniziale del walk forward)
- 24 versione (step finale del walk forward) quando possibile, alrimenti lo step che massimizza il training set

Precision - minima dimensione training set

#TrainingRelease	%Defective in training	%Defective in testing	Classifier	Balancing	Feature Selection	Sensitivity	TP	FP	TN	FN	Precision	Recall	AUC	Kappa
1	0,889	0,836	NaiveBayes	OVERSAMPLING	BEST_FIRST	NONE	34	0	11	22	1	0,607	0,871	0,337
1	0,889	0,836	RandomForest	UNDERSAMPLING	BEST_FIRST	NONE	34	0	11	22	1	0,607	0,842	0,337

I risultati sono interessanti perché anche applicando un bilanciamento di tipo *UNDERSAMPLING* si ottiene un'alta Precision.

#TrainingRelease	%Defective in training	%Defective in testing	Classifier	Balancing	Feature Selection	Sensitivity	TP	FP	TN	FN	Precision	Recall	AUC	Kappa
10	0,682	0,75	NaiveBayes	NONE	NONE	NONE	50	0	70	160	1	0,238	0,772	0,135
10	0,682	0,75	NaiveBayes	OVERSAMPLING	NONE	NONE	52	0	70	158	1	0,248	0,769	0,141
10	0,682	0,75	NaiveBayes	UNDERSAMPLING	NONE	NONE	50	0	70	160	1	0,238	0,77	0,135
10	0,682	0,75	NaiveBayes	SMOTE	NONE	NONE	48	0	70	162	1	0,229	0,771	0,129
10	0,682	0,75	NaiveBayes	OVERSAMPLING	NONE	SENSITIVE_LEARNING	45	0	70	165	1	0,214	0,763	0,12
10	0,815	0,88	NaiveBayes	OVERSAMPLING	BEST_FIRST	NONE	10	0	10	63	1	0,137	0,846	0,037
10	0,815	0,88	NaiveBayes	UNDERSAMPLING	BEST_FIRST	NONE	23	0	10	50	1	0,315	0,861	0,1
10	0,815	0,88	RandomForest	UNDERSAMPLING	BEST_FIRST	SENSITIVE_LEARNING	58	1	9	15	0,983	0,795	0,847	0,433
10	0,815	0,88	Ibk	UNDERSAMPLING	BEST_FIRST	SENSITIVE_LEARNING	54	1	9	19	0,982	0,74	0,82	0,36

La release massima in cui si ha Precision pari a 1 è la decima e si ha solo per il calssificatore NaiveBayes. Tuttavia, a parità di dati, paragonando tutte le metriche di questo classificatore con quelle degli altri modelli, si nota che questi ultimi hanno una precision leggermente più bassa ma migliorano decisamente le altre metriche.

Recall - minima dimensione training set

#TrainingRelease	%Defective in training	%Defective in testing	Classifier	Balancing	Feature Selection	Sensitivity	TP	FP	TN	FN	Precision	Recall	AUC	Kappa
1	0,727	0,807	RandomForest	NONE	NONE	SENSITIVE_LEARNING	209	41	9	0	0,836	1	0,672	0,262
1	0,727	0,807	NaiveBayes	OVERSAMPLING	NONE	SENSITIVE_LEARNING	209	50	0	0	0,807	1	0,493	0
1	0,727	0,807	NaiveBayes	SMOTE	NONE	SENSITIVE_LEARNING	209	50	0	0	0,807	1	0,495	0
1	0,889	0,836	Ibk	NONE	BEST_FIRST	NONE	56	11	0	0	0,836	1	0,586	0
1	0,889	0,836	RandomForest	NONE	BEST_FIRST	SENSITIVE_THRESHOLD	56	11	0	0	0,836	1	0,5	0
1	0,889	0,836	NaiveBayes	NONE	BEST_FIRST	SENSITIVE_THRESHOLD	56	11	0	0	0,836	1	0,5	0
1	0,889	0,836	Ibk	NONE	BEST_FIRST	SENSITIVE_THRESHOLD	56	11	0	0	0,836	1	0,5	0
1	0,889	0,836	Ibk	OVERSAMPLING	BEST_FIRST	SENSITIVE_THRESHOLD	56	11	0	0	0,836	1	0,5	0
1	0,889	0,836	RandomForest	UNDERSAMPLING	BEST_FIRST	SENSITIVE_THRESHOLD	56	11	0	0	0,836	1	0,5	0
1	0,889	0,836	NaiveBayes	UNDERSAMPLING	BEST_FIRST	SENSITIVE_THRESHOLD	56	11	0	0	0,836	1	0,5	0
1	0,889	0,836	Ibk	UNDERSAMPLING	BEST_FIRST	SENSITIVE_THRESHOLD	56	11	0	0	0,836	1	0,5	0
1	0,889	0,836	RandomForest	SMOTE	BEST_FIRST	SENSITIVE_THRESHOLD	56	11	0	0	0,836	1	0,5	0
1	0,889	0,836	NaiveBayes	SMOTE	BEST_FIRST	SENSITIVE_THRESHOLD	56	11	0	0	0,836	1	0,5	0
1	0,889	0,836	RandomForest	NONE	BEST_FIRST	SENSITIVE_LEARNING	56	11	0	0	0,836	1	0,724	0
1	0,889	0,836	NaiveBayes	NONE	BEST_FIRST	SENSITIVE_LEARNING	56	11	0	0	0,836	1	0,866	0
1	0,889	0,836	Ibk	NONE	BEST_FIRST	SENSITIVE_LEARNING	56	11	0	0	0,836	1	0,586	0
1	0,889	0,836	NaiveBayes	OVERSAMPLING	BEST_FIRST	SENSITIVE_LEARNING	56	11	0	0	0,836	1	0,473	0

Sebbene ci siano 17 casi in cui la recall è 1 con un training set minimale, l'unico caso degno di nota è il primo: gli altri, infatti, presentano tutti il valore di Kappa pari a 0.

Nel primo caso notiamo il *sensitive learning* come tecnica di sensitivity: questo fatto, insieme alla alta percentuale di difetti nel testing, giustifica un numero di *false negative* pari a 0. Inoltre, queste osservazioni giustificano anche un valore di precision così alto.

Recall - massima dimensione training set

#TrainingRelease	%Defective in training	%Defective in testing	Classifier	Balancing	Feature Selection	Sensitivity	TP	FP	TN	FN	Precision	Recall	AUC	Kappa
24	0,394	0,198	RandomForest	NONE	NONE	SENSITIVE_THRESHOLD	173	188	511	0	0,479	1	0,866	0,519
24	0,394	0,198	RandomForest	UNDERSAMPLING	NONE	SENSITIVE_THRESHOLD	173	250	449	0	0,409	1	0,821	0,416
24	0,394	0,198	RandomForest	SMOTE	NONE	SENSITIVE_THRESHOLD	173	192	507	0	0,474	1	0,863	0,512
24	0,394	0,198	NaiveBayes	SMOTE	NONE	SENSITIVE_THRESHOLD	173	699	0	0	0,198	1	0,5	0
24	0,394	0,198	RandomForest	NONE	NONE	SENSITIVE_LEARNING	173	185	514	0	0,483	1	0,973	0,524
24	0,545	0,324	RandomForest	NONE	BEST_FIRST	SENSITIVE_THRESHOLD	47	40	58	0	0,54	1	0,796	0,485
24	0,545	0,324	NaiveBayes	NONE	BEST_FIRST	SENSITIVE_THRESHOLD	47	98	0	0	0,324	1	0,5	0
24	0,545	0,324	NaiveBayes	UNDERSAMPLING	BEST_FIRST	SENSITIVE_THRESHOLD	47	98	0	0	0,324	1	0,5	0
24	0,545	0,324	RandomForest	NONE	BEST_FIRST	SENSITIVE_LEARNING	47	47	51	0	0,5	1	0,88	0,413
24	0,545	0,324	NaiveBayes	NONE	BEST_FIRST	SENSITIVE_LEARNING	47	98	0	0	0,324	1	0,809	0

Nel caso di training set massimale, si può notare che una recall pari a 1 si ha solo se utilizzate tecniche di sensitivity.

AUC

#TrainingRelease	%Defective in training	%Defective in testing	Classifier	Balancing	Feature Selection	Sensitivity	TP	FP	TN	FN	Precision	Recall	AUC	Kappa
5	0,845	0,962	NaiveBayes	OVERSAMPLING	BEST_FIRST	SENSITIVE_LEARNING	100	4	0	0	0,962	1	0,985	0
24	0,394	0,198	RandomForest	NONE	NONE	NONE	133	29	670	40	0,821	0,769	0,978	0,745
24	0,394	0,198	Ibk	NONE	NONE	NONE	166	60	639	7	0,735	0,96	0,958	0,783
24	0,394	0,198	Ibk	SMOTE	NONE	NONE	166	60	639	7	0,735	0,96	0,958	0,783
24	0,394	0,198	Ibk	NONE	NONE	SENSITIVE_LEARNING	166	60	639	7	0,735	0,96	0,958	0,783
24	0,545	0,324	RandomForest	NONE	BEST_FIRST	NONE	33	17	81	14	0,66	0,702	0,882	0,52
24	0,545	0,324	Ibk	NONE	BEST_FIRST	NONE	39	26	72	8	0,6	0,83	0,808	0,513
24	0,545	0,324	Ibk	SMOTE	BEST_FIRST	NONE	38	24	74	9	0,613	0,809	0,811	0,52
24	0,545	0,324	Ibk	NONE	BEST_FIRST	SENSITIVE_LEARNING	39	26	72	8	0,6	0,83	0,806	0,513

- **Naive Bayes**: il valore massimo di AUC si ha quando il classificatore assegna sempre il valore *Yes*.
- **Ibk**: le tre configurazioni producono la stessa accuratezza, e in nessuna di queste è usata la features selection. È strano che non ci siano differenze nel caso di *SENSITIVE LEARNING* così come è strano che la tecnina di Best First porti a una diminuzione di circa il 10% su tutte.
- **Random Forest**: presenta una misura di AUC molto alta nonostante non sia applicata nessuna tecnica

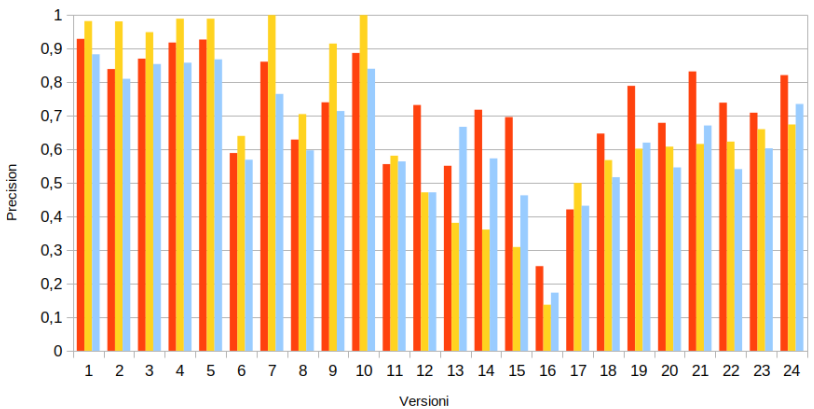
Kappa

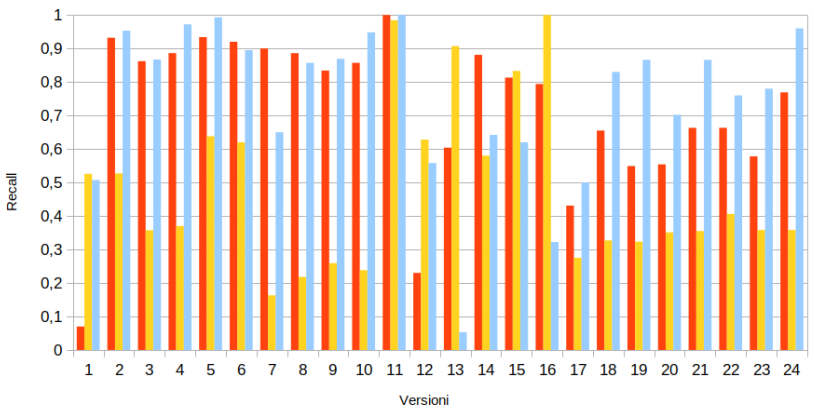
#TrainingRelease	%Defective in training	%Defective in testing	Classifier	Balancing	Feature Selection	Sensitivity	TP	FP	TN	FN	Precision	Recall	AUC	Kappa
24	0,394	0,198	RandomForest	SMOTE	NONE	NONE	140	38	661	33	0,787	0,809	0,977	0,747
24	0,394	0,198	Ibk	NONE	NONE	NONE	166	60	639	7	0,735	0,96	0,958	0,783
24	0,394	0,198	Ibk	SMOTE	NONE	NONE	166	60	639	7	0,735	0,96	0,958	0,783
24	0,394	0,198	Ibk	NONE	NONE	SENSITIVE_THRESHOLD	166	60	639	7	0,735	0,96	0,937	0,783
24	0,394	0,198	Ibk	SMOTE	NONE	SENSITIVE_THRESHOLD	166	60	639	7	0,735	0,96	0,937	0,783
24	0,394	0,198	Ibk	NONE	NONE	SENSITIVE_LEARNING	166	60	639	7	0,735	0,96	0,958	0,783
3	0,802	0,929	NaiveBayes	OVERSAMPLING	BEST_FIRST	SENSITIVE_LEARNING	24	0	2	2	1	0,923	0,971	0,632

- **Naive Bayes**: il massimo valore di Kappa per questo classificatore è del 63%, e si verifica in presenza di un testing set molto semplice. In più, per ottenere questo valore è necessario applicare tattiche di ogni tipo
- Per gli altri classificatori il valore massimo è raggiunto in presenza della massima quantità dei dati nel training set con valori intorno al 75%.

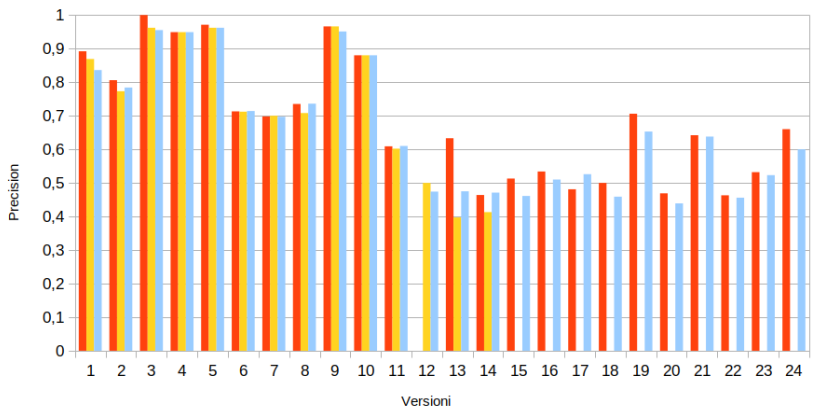
Confronto generale classificatori - Feature selection

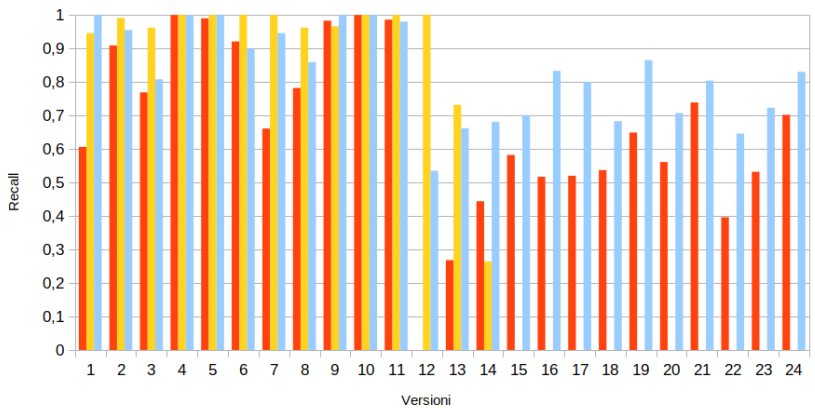
No feature selection





feature selection

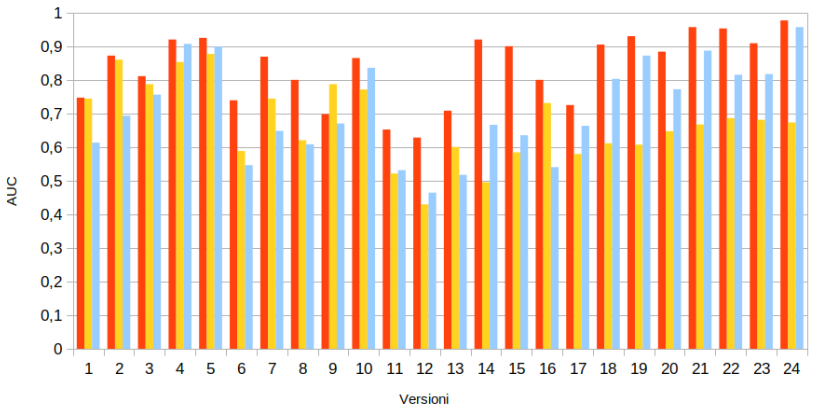


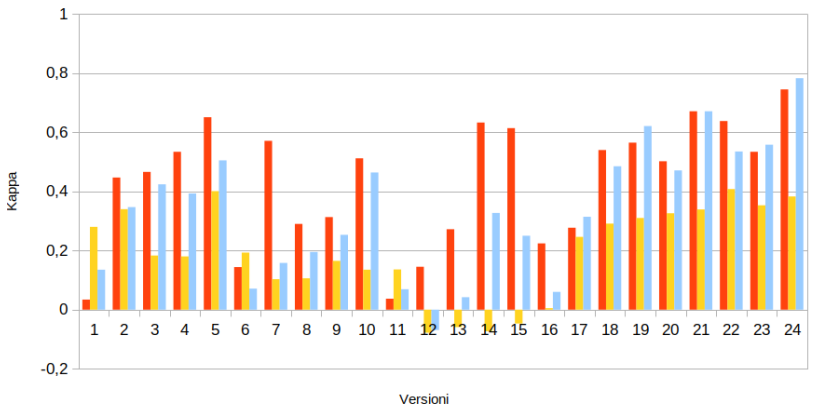


RandomForest Naive Bayes Ibk

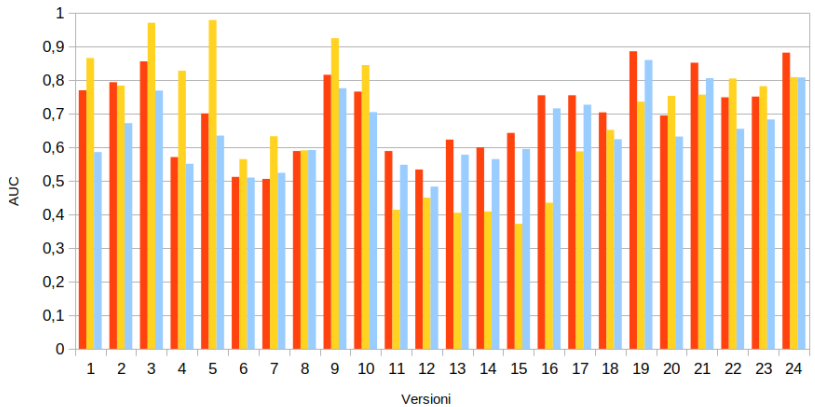
Confronto generale classificatori - Feature selection

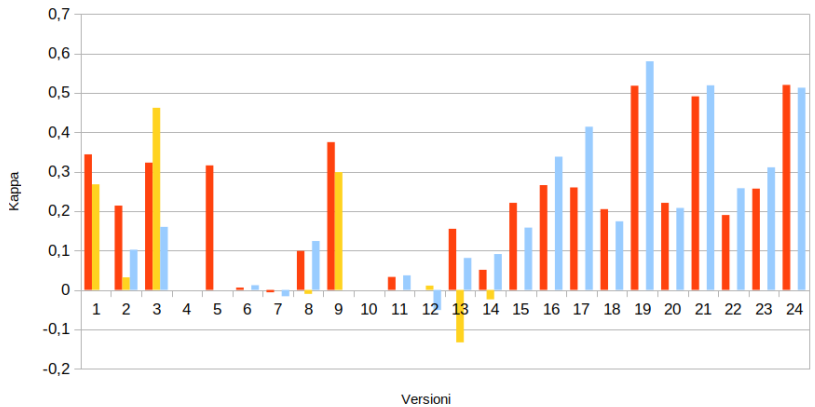
No feature selection





feature selection



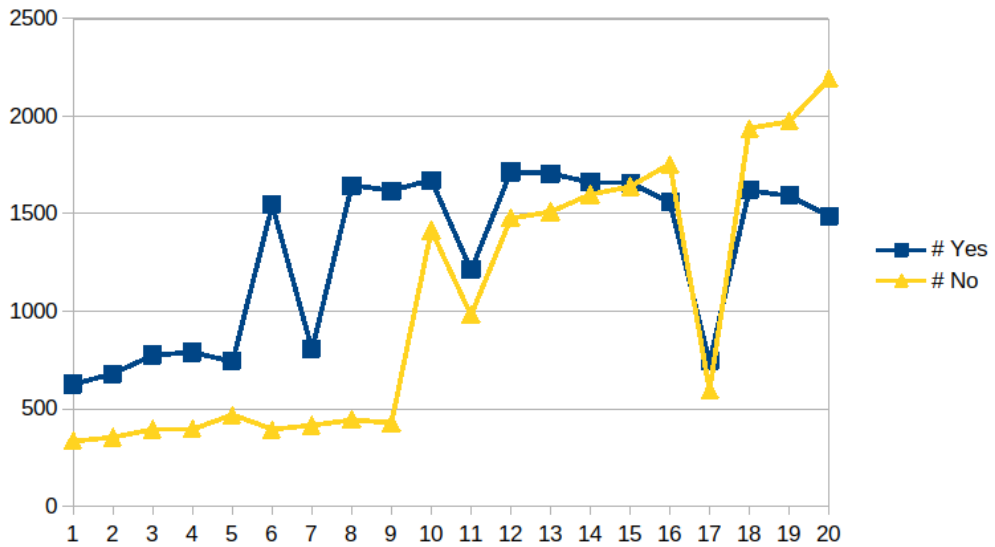


RandomForest Naive Bayes Ibk

Proportion classification

Il numero di `Yes` è quasi sempre superiore a quelli di `No`.

Version	# Yes	# No	% Yes	% No
1	625	336	0,65	0,35
2	679	353	0,66	0,34
3	776	395	0,66	0,34
4	789	397	0,67	0,33
5	743	467	0,61	0,39
6	1546	392	0,8	0,2
7	804	415	0,66	0,34
8	1645	445	0,79	0,21
9	1617	426	0,79	0,21
10	1671	1415	0,54	0,46
11	1212	981	0,55	0,45
12	1713	1479	0,54	0,46
13	1706	1510	0,53	0,47
14	1663	1598	0,51	0,49
15	1657	1640	0,5	0,5
16	1559	1752	0,47	0,53
17	744	595	0,56	0,44
18	1620	1937	0,46	0,54
19	1593	1975	0,45	0,55
20	1487	2193	0,4	0,6



È interessante analizzare i picci e le cadute nel grafico precedenti; vengono riportate delle tabelle di supporto:

Version	# files
6	1938
7	1219
10	3086
11	2193
16	3311
17	1339

Version	nfix tot
10	4409

Massimi valori per i classificatori

Classificatore	Max Precision	Occorrenze Max Precision
RandomForest	1	1
NaiveBayes	0,995	1
Ibk	1	1

Classificatore	Max Recall	Occorrenze Max Recall
RandomForest	1	53
NaiveBayes	1	56
Ibk	0,996	1

Classificatore	Max AUC	Occorrenze Max AUC
RandomForest	0,921	1
NaiveBayes	0,843	1
Ibk	0,906	1

Classificatore	Max Kappa	Occorrenze Max Kappa
RandomForest	0,576	1
NaiveBayes	0,473	1
Ibk	0,543	1

Per la Recall, si analizzeranno i casi in cui la dimensione del training set è massima o minima.

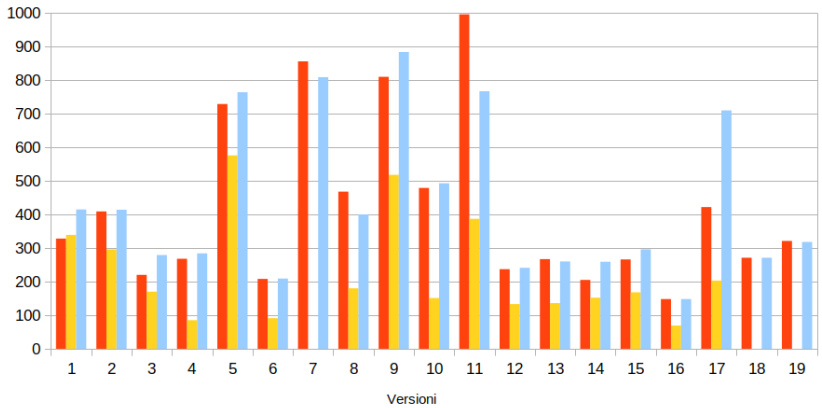
Precision

#TrainingRelease	%Defective in training	%Defective in testing	Classifier	Balancing	Feature Selection	Sensitivity	TP	FP	TN	FN	Precision	Recall	AUC	Kappa
1	0,66	0,707	Ibk	UNDERSAMPLING	NONE	SENSITIVE_LEARNING	10	0	232	549	1	0,018	0,509	0,011
8	0,838	0,864	RandomForest	UNDERSAMPLING	BEST_FIRST	NONE	425	0	99	204	1	0,676	0,874	0,362
8	0,838	0,864	NaiveBayes	UNDERSAMPLING	BEST_FIRST	SENSITIVE_LEARNING	161	1	98	468	0,994	0,256	0,838	0,082

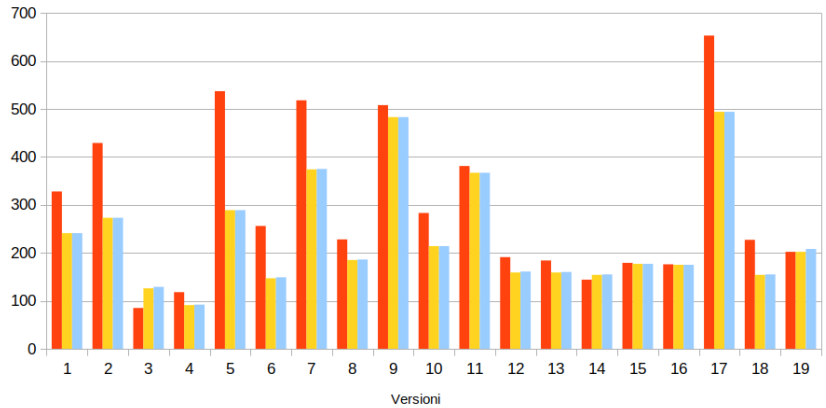
I risultati sono particolari: presentano una precision pari (o quasi) a 1, ma la recall è bassissima nonostante sia applicata la tecnica di *sensitive learning*. Per questo motivo si è analizzato l'andamento dei falsi negativi nel corso delle versioni, fissando il *balancing* e la *feature selection*.

Andamento falsi negativi

sensitive learning sensitive threshold none



Balancing: `UNDERSAMPLING`
Feature Selection: `BEST_FIRST`
Classificatore: `NaiveBayes`



Balancing: `UNDERSAMPLING`
Feature Selection: `BEST_FIRST`
Classificatore: `Ibk`

Altre configurazioni possibili sul foglio excell...

Recall

#TrainingRelease	%Defective in training	%Defective in testing	Classifier	Balancing	Feature Selection	Sensitivity	TP	FP	TN	FN	Precision	Recall	AUC	Kappa
2	0,682	0,787	RandomForest	NONE	NONE	SENSITIVE_THRESHOLD	554	150	0	0	0,787	1	0,5	0
2	0,682	0,787	RandomForest	NONE	NONE	SENSITIVE_LEARNING	554	150	0	0	0,787	1	0,784	0
2	0,682	0,787	NaiveBayes	SMOTE	NONE	SENSITIVE_LEARNING	552	150	0	2	0,786	0,996	0,62	-0,006
2	0,686	0,787	RandomForest	NONE	BEST_FIRST	SENSITIVE_LEARNING	550	149	0	0	0,787	1	0,779	0

La recall massima raggiunta dai classificatori con poche release è funzione della semplicità del dataset: infatti tutti funzionano come un classificatore random che classifica tutte le istanze come positive. La precision risulta alta, ma coincide con la percentuale di istanze positive nel testing set.

#TrainingRelease	%Defective in training	%Defective in testing	Classifier	Balancing	Feature Selection	Sensitivity	TP	FP	TN	FN	Precision	Recall	AUC	Kappa
19	0,719	0,695	RandomForest	NONE	NONE	SENSITIVE_THRESHOLD	748	326	2	0	0,696	1	0,503	0,008
19	0,719	0,695	RandomForest	SMOTE	NONE	SENSITIVE_THRESHOLD	748	321	7	0	0,7	1	0,511	0,029
19	0,743	0,695	NaiveBayes	NONE	BEST_FIRST	SENSITIVE_THRESHOLD	748	328	0	0	0,695	1	0,5	0
19	0,743	0,695	NaiveBayes	OVERSAMPLING	BEST_FIRST	SENSITIVE_THRESHOLD	748	328	0	0	0,695	1	0,5	0
19	0,743	0,695	NaiveBayes	UNDERSAMPLING	BEST_FIRST	SENSITIVE_THRESHOLD	748	328	0	0	0,695	1	0,5	0
19	0,743	0,695	RandomForest	SMOTE	BEST_FIRST	SENSITIVE_THRESHOLD	748	318	10	0	0,702	1	0,515	0,042
19	0,743	0,695	NaiveBayes	SMOTE	BEST_FIRST	SENSITIVE_THRESHOLD	748	328	0	0	0,695	1	0,5	0
19	0,743	0,695	NaiveBayes	NONE	BEST_FIRST	SENSITIVE_LEARNING	748	328	0	0	0,695	1	0,775	0
19	0,743	0,695	NaiveBayes	SMOTE	BEST_FIRST	SENSITIVE_LEARNING	748	328	0	0	0,695	1	0,781	0
9	0,809	0,616	Ibk	SMOTE	NONE	SENSITIVE_LEARNING	1465	912	4	6	0,616	0,996	0,538	0

Anche nel caso di training set massimale le cose non migliorano: recall massima, ma kappa pari a 0 sempre. È interessante come valori di recall così alta si verificano solo in presenza di tecniche di *sensibility*.

AUC

#TrainingRelease	%Defective in training	%Defective in testing	Classifier	Balancing	Feature Selection	Sensitivity	TP	FP	TN	FN	Precision	Recall	AUC	Kappa
8	0,802	0,864	RandomForest	NONE	NONE	NONE	580	44	55	49	0,929	0,922	0,921	0,468
8	0,838	0,864	Ibk	OVERSAMPLING	BEST_FIRST	NONE	518	3	96	111	0,994	0,824	0,906	0,543
8	0,838	0,864	NaiveBayes	OVERSAMPLING	BEST_FIRST	SENSITIVE_LEARNING	183	1	98	446	0,995	0,291	0,843	0,097

Il valore massimo di AUC si ha, per ogni classificatore, in corrispondenza di un training set composto da 8 versioni.

- **Random Forest e Ibk** presentano delle ottime valutazioni anche per precision e recall, ma il valore di kappa non è molto alto
- **NaiveBayes**: presenta un valore kappa pari pressocché a 0, poco interessante

#TrainingRelease	%Defective in training	%Defective in testing	Classifier	Balancing	Feature Selection	Sensitivity	TP	FP	TN	FN	Precision	Recall	AUC	Kappa
19	0,719	0,695	RandomForest	NONE	NONE	NONE	659	183	145	89	0,783	0,881	0,827	0,351
19	0,719	0,695	Ibk	UNDERSAMPLING	NONE	NONE	559	117	211	189	0,827	0,747	0,759	0,368
19	0,719	0,695	NaiveBayes	NONE	NONE	NONE	462	39	289	286	0,922	0,618	0,786	0,412
19	0,719	0,695	NaiveBayes	NONE	NONE	SENSITIVE_LEARNING	746	327	1	2	0,695	0,997	0,786	0,001
19	0,719	0,695	NaiveBayes	OVERSAMPLING	NONE	SENSITIVE_LEARNING	334	24	304	414	0,933	0,447	0,786	0,28

Le stesse caratteristiche si verificano quando il training set è massimale.

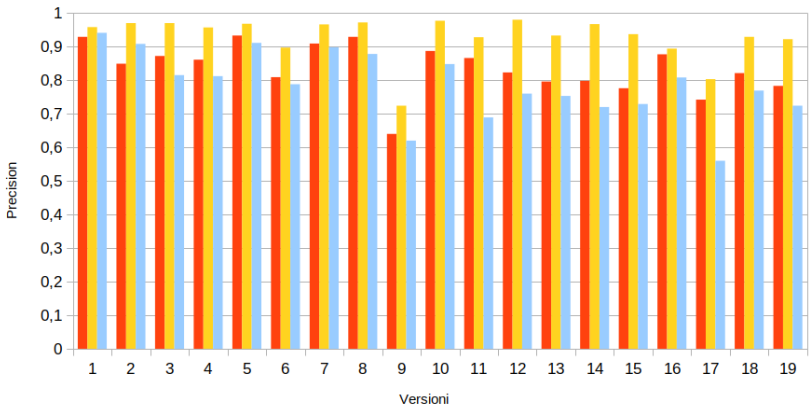
Kappa

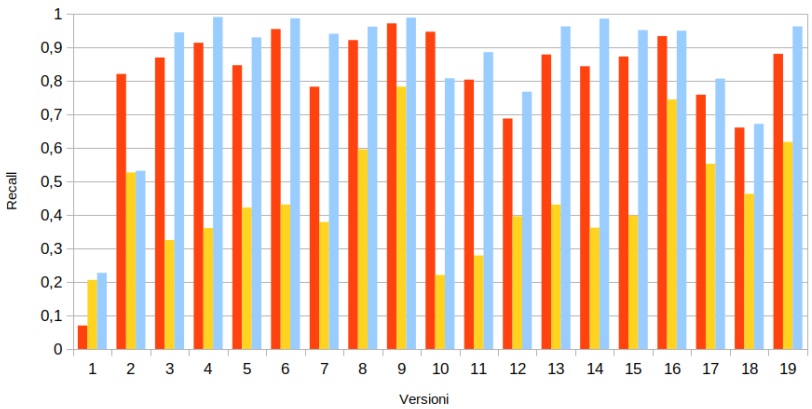
#TrainingRelease	%Defective in training	%Defective in testing	Classifier	Balancing	Feature Selection	Sensitivity	TP	FP	TN	FN	Precision	Recall	AUC	Kappa
8	0,838	0,864	Ibk	OVERSAMPLING	BEST_FIRST	NONE	518	3	96	111	0,994	0,824	0,906	0,543
16	0,78	0,797	RandomForest	UNDERSAMPLING	BEST_FIRST	NONE	584	39	133	90	0,937	0,866	0,888	0,576
11	0,795	0,684	NaiveBayes	UNDERSAMPLING	BEST_FIRST	SENSITIVE_THRESHOLD	1029	133	520	387	0,886	0,727	0,762	0,473

Come si è visto già nella slide che riassume i *massimi valori dei classificatori*, il valore kappa massimo è comunque molto basso. Il valore massimo tra i tre classificatori è quello del Random Forest che, tra i tre casi, ha la massima dimensione del training set.

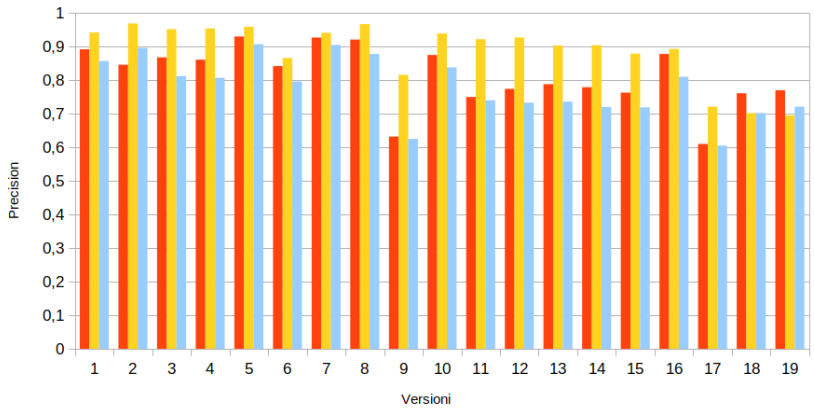
Confronto generale classificatori - Feature selection

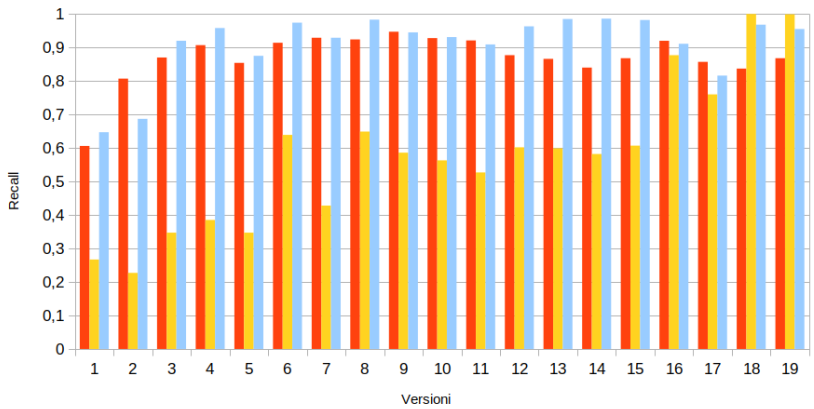
No feature selection





feature selection

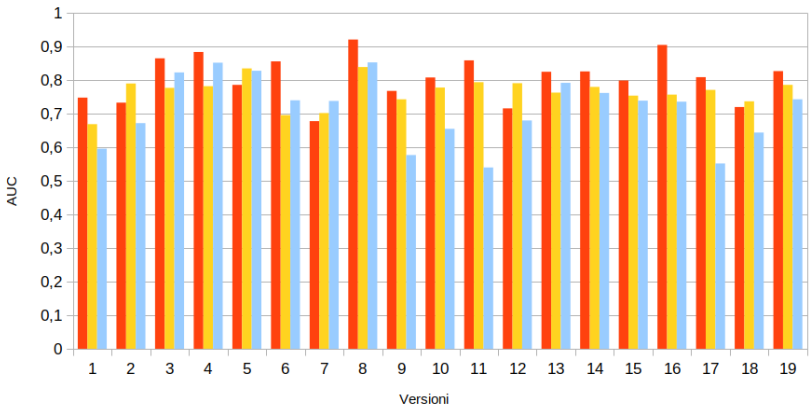


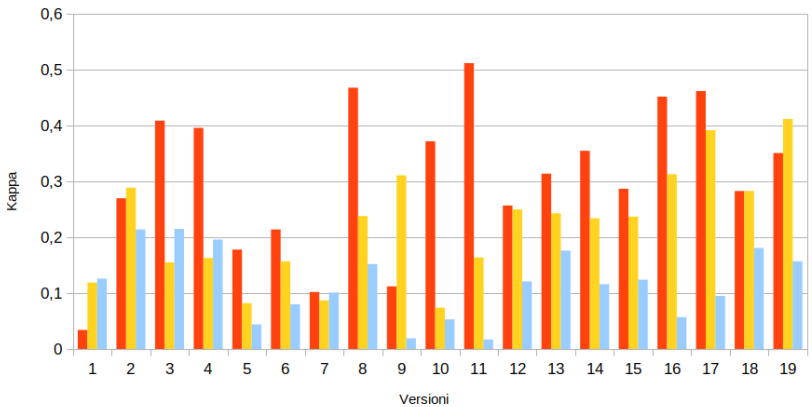


RandomForest Naive Bayes Ibk

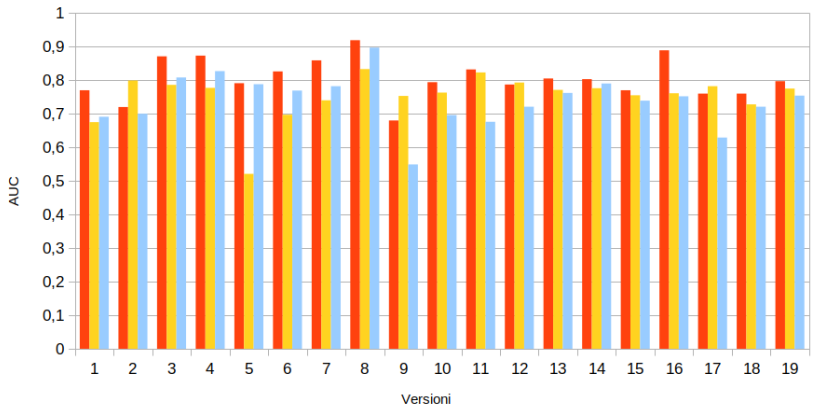
Confronto generale classificatori - Feature selection

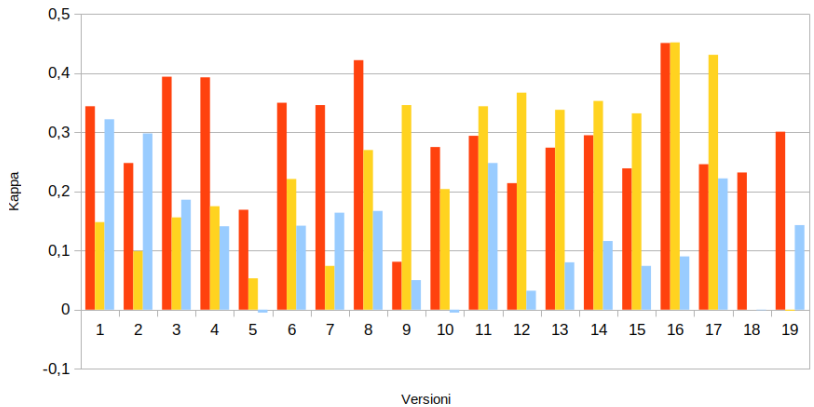
No feature selection





feature selection





RandomForest Naive Bayes Ibk