Laborator 3.2 - Tehnologii NoSQL

Metode probabilistice pentru clasificare - Regresia logistica

Gheorghe Cosmin Silaghi

Universitatea Babeș-Bolyai

March 31, 2023

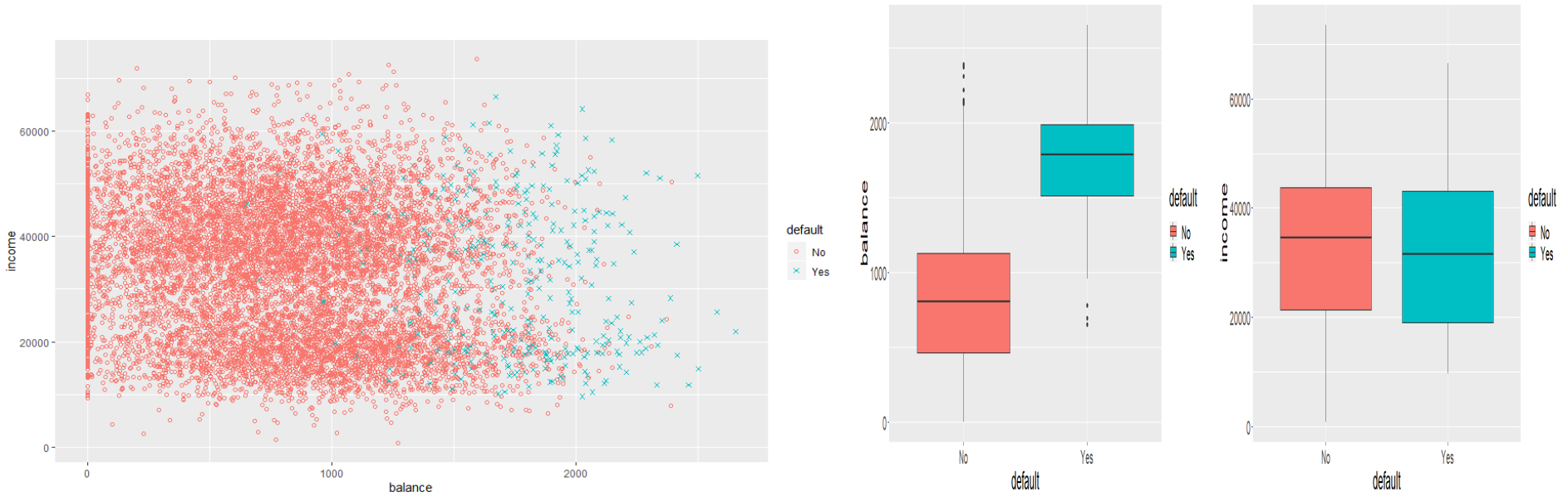
Intrebare de cercetare

- Dorim sa prezicem daca un posesor al unui card de credit va fi in situatia sa nu isi poata plati creditul luat (Default == yes) pe baza venitului anual si a balantei lunare a cardului de credit.
- Mai mult dorim sa aflam daca situatia persoanei (daca persoana este un student) influenteaza sau nu posibilitatea de Default.

Intrebare de cercetare

- Dorim sa prezicem daca un posesor al unui card de credit va fi in situatia sa nu isi poata plati creditul luat (Default == yes) pe baza venitului anual si a balantei lunare a cardului de credit.
- Mai mult dorim sa aflam daca situatia persoanei (daca persoana este un student) influenteaza sau nu posibilitatea de Default.

Pas 1 - Incarcarea si vizualizarea datelor



Regresia logistica

- modeleaza probabilitatea unei instante de a apartie unei *categorii* **particulare** *a clasei tinta*

$$p(balance) \overset{Not}{=} Pr(default = yes|balance) \in [0, 1] \tag{1}$$

Regresia logistica

- modeleaza probabilitatea unei instante de a apartie unei *categorii* **particulare** *a clasei tinta*

$$p(balance) \overset{Not}{=} Pr(default = yes|balance) \in [0, 1] \tag{1}$$

- Prezicem `default = yes` daca $p(balance) > threshold$

Regresia logistica

- modeleaza probabilitatea unei instante de a apartie unei *categorii* **particulare** *a clasei tinta*

$$p(balance) \overset{Not}{=} Pr(default = yes|balance) \in [0, 1] \tag{1}$$

- Prezicem `default = yes` daca $p(balance) > threshold$

Functia logistica

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}} \tag{2}$$

- $p(X)$ ia valori exclusiv intre $(0, 1)$ si va fi o curba in forma de S

Regresia logistica II

Odds

$$\frac{p(X)}{1-p(X)} = e^{\beta_0+\beta_1 X} \tag{3}$$

- valoare apropiata de zero indica o probabilitate foarte mica pentru default
- valoare apropiata de infinit indica o probabilitate foarte mare pentru default
- utilizat in jocurile de noroc (betting)

Regresia logistica II

Odds

$$\frac{p(X)}{1-p(X)} = e^{\beta_0+\beta_1 X} \tag{3}$$

- valoare apropiata de zero indica o probabilitate foarte mica pentru default
- valoare apropiata de infinit indica o probabilitate foarte mare pentru default
- utilizat in jocurile de noroc (betting)

Log-odds sau *logit*

$$\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X \tag{4}$$

- daca $\beta_1 > 0 \Rightarrow$ cresterea lui X este asociata cu o crestere a lui $p(X)$
- daca $\beta_1 < 0 \Rightarrow$ crescator X este asociata cu o descrestere a lui $p(X)$

Metoda verosimilitarii maxime

incercam sa gasim valorile $\hat{\beta}_0$ si $\hat{\beta}_1$ astfel incat, inlocuite in functia logistica, aceasta calculeaza valori aproape de 1 pentru instantele care apartin categoriei respective, si calculeaza valori aproape de zero pentru instantele care nu apartin acelei categorii

Metoda verosimilitarii maxime

incercam sa gasim valorile $\hat{\beta}_0$ si $\hat{\beta}_1$ astfel incat, inlocuite in functia logistica, aceasta calculeaza valori aproape de 1 pentru instantele care apartin categoriei respective, si calculeaza valori aproape de zero pentru instantele care nu apartin acelei categorii

Functia de verosimilitate (likelihood)

$$\ell(\beta_0, \beta_1) = \prod_{i:y_i=1} p(x_i) \prod_{i':y_{i'}=0} (1 - p(x_{i'}))$$

- β_0 si β_1 sunt alesi astfel incat sa *maximizeze* functia de verosimilitate

Metoda verosimilitarii maxime

incercam sa gasim valorile $\hat{\beta}_0$ si $\hat{\beta}_1$ astfel incat, inlocuite in functia logistica, aceasta calculeaza valori aproape de 1 pentru instantele care apartin categoriei respective, si calculeaza valori aproape de zero pentru instantele care nu apartin acelei categorii

Functia de verosimilitate (likelihood)

$$\ell(\beta_0, \beta_1) = \prod_{i:y_i=1} p(x_i) \prod_{i':y_{i'}=0} (1 - p(x_{i'}))$$

- β_0 si β_1 sunt alesi astfel incat sa *maximizeze* functia de verosimilitate

R code

```
mod <- glm(data = Default, default ~ balance, family = binomial)
summary(mod)
```

	Coeficient	Std. error	Z-statistic	p-value
Intercept	-10.6513	0.3612	-29.5	< 0.0001
balance	0.0055	0.0002	24.9	< 0.0001

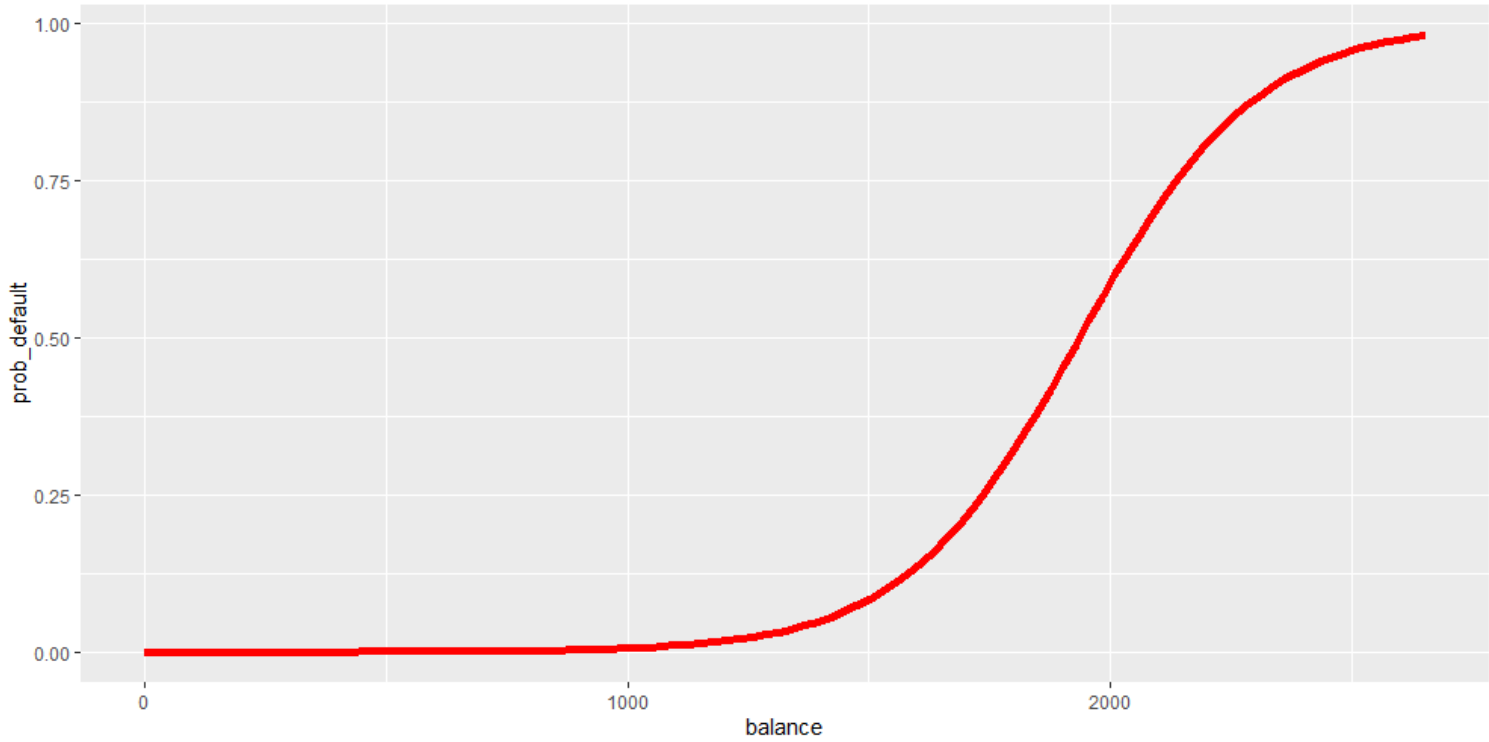
Realizarea predictiilor

R code

```
nd <- tribble(~balance, 1000, 2000)
predicted <- predict(mod, newdata = nd, type =
"response")
```

Predictii

$$Pr(default = Yes|balance = 1000) = 0.0058$$
$$Pr(default = Yes|balance = 2000) = 0.5857$$



Variabile nominale in regresia logistica

- In regresia logistica putem avea variabile binare (calitative) ca si predictori (e.g. variabila student)

Cod R

```
mod_student <- glm(data = Default, default ~ student, family = binomial)
```

	Coeficient	Std. error	Z-statistic	p-value
`Intercept`	-3.5041	0.0707	-49.55	< 0.0001
`student:Yes`	0.4049	0.1150	3.52	0.00043

	value
$\hat{Pr}(default = Yes \vert student = Yes)$	0.0431
$\hat{Pr}(default = Yes \vert student = No)$	0.0292

Regresie logistica multipla

- mai multe variabile independente in regresie

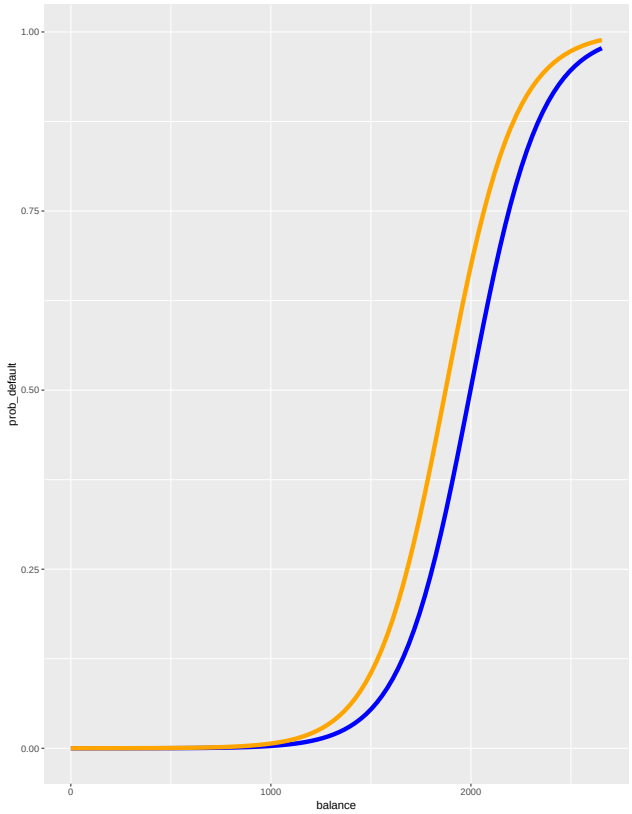
Cod R

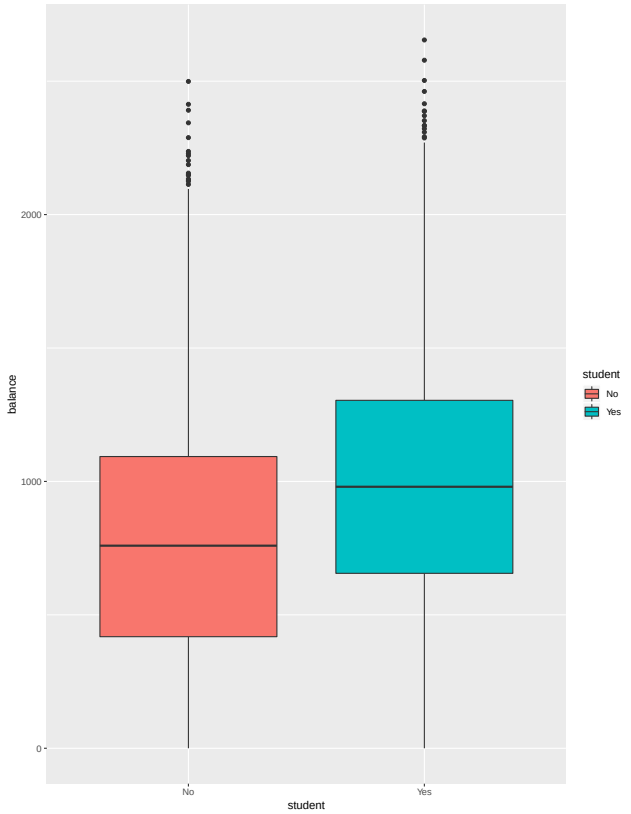
```
mod_all <- glm(data = Default, default ~ balance + income + student, family = binomial) summary(mod_all)
```

	Coeficient	Std. error	Z-statistic	p-value
`Intercept`	-10.869	0.4923	-22.08	< 0.0001
`balance`	0.0057	0.0002	24.738	< 0.0001
`income`	3.03×10^{-6}	0.000008	0.37	0.7115
`student:Yes`	-0.6467	0.2363	-2.738	0.0062

Confounding

- Atunci cand este inclusa in model, o variabila *confounding* schimba sensul predictiei variabilei dependente,
- Modelul fara informatii cu privire la credit card prezice faptul ca studentii sunt predispusi la *default*
- Cand este inclusa in model si balanta cardului de credit, descoperim faptul ca un student devine un client mai putin riscant, decat o persoana care nu este student





Realizarea predictiilor si matricea de confuzie

- trebuie aplicat setupul complet de ML pentru antrenare si testare, inclusiv selectarea unei metode de validare a rezultatelor
- Pentru obtinerea celui mai bun model se poate folosi direct GLM sau utiliza package-ul Caret (pentru cross-validare si cautare parametri)

Cod R

```
pred_test <- predict(mod_balance_student_train, newdata = test, type = "response")
table(pred_test > 0.5, test$default)
```

Threshold 0.5		True values	
		No	Yes
Predicted values	No	2878	78
	Yes	19	25

Threshold 0.2		True values	
		No	Yes
Predicted values	No	2815	49
	Yes	82	54

- identificam doar 25 de defaulters din cele peste 100 persoane in setul de test: $\Rightarrow$ specificitatea este foarte mica (24.27%)
- putem creste numarul defaulters identificati daca scadem pragul la 0.2 $\Rightarrow$ specificitatea creste la (52.42%)
- acest lucru se intampla cu costul cresterii numarului de *false positives*
- prin modificarea succesiva a pragului se poate obtine curba ROC