Istraživanje podataka 1 - vežbe 6, 2020.

1 Naivni Bajesovski klasifikatori

1.1 Uslovna verovatnoća i Bajesova teorema

Neka su A i C dva događaja. Verovatnoću da se zajedno dese događaj A i događaj C označavmo sa $P(A, C)$.
Uslovnu verovatnoću da se desi događaj C, ako se desio događaj A označavamo sa $P(X|A)$ i računamo

$$P(C|A) = \frac{P(A, C)}{P(A)}$$

gde je P(A) verovatnoća da se desi događaj A.
Verovatnoću da se zajedno dese događaj A i događaj C možemo računati i sa

$$P(A, C) = P(C|A) * P(A)$$

kao i sa

$$P(A, C) = P(A|C) * P(C)$$

te važi

$$P(C|A) * P(A) = P(A|C) * P(C)$$

te $P(C|A)$ može i da se računa sa

$$P(C|A) = \frac{P(A|C) * P(C)}{P(A)}$$

što je **Bajesova teorema**.

1.2 Bajesovski klasifikatori

Bajesovski klasifikatori koriste Bajesovu teoremu za predviđanje klase test instance. Neka je test instanca opisana sa $A = (A_1, A_2, ..., A_n)$. Da bi se dodelila klasa test instanci, potrebno je pronaći klasu C koja ima najveću uslovnu verovatnoću $P(C|(A_1, A_2, ..., A_n))$.

Naivni Bajesovski klasifikator uzima pretpostavku o nezavisnosti između atributa $A_1, A_2, ..., A_n$, te se uslovna verovatnoća za klasu C računa sa

$$P(C|A_1, A_2, ..., A_n) = \frac{\prod_{i=1}^{n} P(A_i|C) * P(C)}{P(A)}$$

Kako je verovatnoća pojavljivanja instance A ista pri računanju uslovne verovatnoće za svaku klasu, test instanci se može dodeliti klasa $\widehat{C}$ računanjem

$$\widehat{C} = \arg\max_{C} \prod_{i=1}^{n} P(A_i|C) * P(C)$$

1.3 Zadaci

1. Dat je skup podataka:

Record	A	B	C	Class
1	0	0	0	+
2	0	0	1	−
3	0	1	1	−
4	0	1	1	−
5	0	0	1	+
6	1	0	1	+
7	1	0	1	−
8	1	0	1	−
9	1	1	1	+
10	1	0	1	+

Predvideti oznaku klase za test instancu $X = (A = 0, B = 1, C = 0)$ koristeći naivan Bajesov pristup.

Rešenje

Da bismo odredili klasu test instance X, računamo uslovne verovatnoće $P(+|X)$ i $P(-|X)$.

$P(+|X) = \frac{P(A=0|+)*P(B=1|+)*P(C=0|+)*P(+)}{P(X)}$

$P(A = 0|+)$ je verovatnoća da instance koje pripadaju klasi + imaju vrednost 0 u atributu A. U trening skupu vidimo da postoji 5 instanci koje pripadaju klasi +, a za 2 od tih 5 instanci važi $A = 0$, pa je verovatnoća $\frac{2}{5}$.

$P(B = 1|+)$ je verovatnoća da instance koje pripadaju klasi + imaju vrednost 1 u atributu B. U trening skupu vidimo da postoji 5 instanci koje pripadaju klasi +, a za 1 od tih 5 instanci važi $B = 1$, pa je verovatnoća $\frac{1}{5}$.

$P(C = 0|+)$ je verovatnoća da instance koje pripadaju klasi + imaju vrednost 0 u atributu C. U trening skupu vidimo da postoji 5 instanci koje pripadaju klasi +, a za 1 od tih 5 instanci važi $C = 0$, pa je verovatnoća $\frac{1}{5}$.

$P(+)$ je verovatnoća da intsanca u trening skupu pripada klasi +. Kako 5 od 10 instanci u trening skupu pripada klasi +, $P(+) = \frac{1}{2}$.

$P(+|X) = \frac{P(A=0|+)*P(B=1|+)*P(C=0|+)*P(+)}{P(X)} = \frac{\frac{2}{5}*\frac{1}{5}*\frac{1}{5}*\frac{1}{2}}{P(X)} = \frac{\frac{1}{5^3}}{P(X)}$

$P(-|X) = \frac{P(A=0|-)*P(B=1|-)*P(C=0|-)*P(-)}{P(X)}$

$P(A = 0|-)$ je verovatnoća da instance koje pripadaju klasi $-$ imaju vrednost 0 u atributu A. U trening skupu vidimo da postoji 5 instanci koje pripadaju klasi $-$, a za 3 od tih 5 instanci važi $A = 0$, pa je verovatnoća $\frac{3}{5}$.

$P(B = 1|-)$ je verovatnoća da instance koje pripadaju klasi $-$ imaju vrednost 1 u atributu B. U trening skupu vidimo da postoji 5 instanci koje pripadaju klasi $-$, a za 2 od tih 5 instanci važi $B = 1$, pa je verovatnoća $\frac{2}{5}$.

$P(C = 0|-)$ je verovatnoća da instance koje pripadaju klasi $-$ imaju vrednost 0 u atributu C. U trening skupu vidimo da postoji 5 instanci koje pripadaju klasi $-$, a za nijednu od tih 5 instanci ne važi $C = 0$, pa je verovatnoća 0.

$P(-)$ je verovatnoća da intsanca u trening skupu pripada klasi $-$. Kako 5 od 10 instanci u trening skupu pripada klasi $-$ $P(-) = \frac{1}{2}$.

$$P(-|X) = \frac{P(A=0|-)*P(B=1|-)*P(C=0|-)*P(-)}{P(X)} = \frac{\frac{3}{5}*\frac{2}{5}*0*\frac{1}{2}}{P(X)} = 0$$

Kako je $P(+|X) > P(-|X)$ test instancu X klasifikujemo klasom $+$.

U ovom primeru možemo da primetimo da ukoliko postoji neka vrednost atributa koja se ne javlja među instancama određene klase u trening skupu, onda će uslovna verovatnoća klase za tu vrednost atributa biti 0, kao u primeru za $P(C = 0|-)$. Zbog toga će i verovatnoća da test instanca pripada toj klasi biti 0, kao u primeru za $P(-|X)$.

U implementacijama naivnog Bajesovog algoritma obično postoji parametar kojim se zadaje verovatnoća za vrednosti atributa koje se ne pojavljuju u trening skupu za neku klasu. Vrednost tog parametra je obično mala (npr. 0,001), i služi da se izbegne problem sa slučajem kada je za neku klasu C verovatnoća $P(C|X) = 0$. Ekstremni slučaj bi bio kada bi za svaku klasu C verovatnoća bila $P(C|X) = 0$.

2. Dati su podaci :

Boja	**Veličina**	**Vrsta**	**Osoba**	**Naduvan**
Žut	Mali	Duguljast	Odrasla	T
Žut	Mali	Duguljast	Dete	T
Žut	Mali	Okrugao	Dete	T
Ljubičast	Veliki	Okrugao	Odrasla	T
Žut	Veliki	Okrugao	Dete	F
Žut	Veliki	Duguljast	Dete	F
Ljubičast	Mali	Okrugao	Dete	F
Ljubičast	Veliki	Duguljast	Odrasla	F

Korišćenjem naivnog Bajesovog algoritma na osnovu prethodno datih podataka klasifikovati sledeće instance i izračunati preciznost. Ciljni atribut je atribut **Naduvan**.

Boja	**Veličina**	**Vrsta**	**Osoba**	**Naduvan**
Ljubičast	Mali	Okrugao	Odrasla	T
Žut	Mali	Okrugao	Odrasla	T
Ljubičast	Veliki	Okrugao	Dete	T
Ljubičast	Veliki	Duguljast	Odrasla	F

Rešenje

Izračunate verovatnoće na osnovu trening skupa koje su potrebne za klasifikaciju test instanci:

X	$P(X\|T)$	$P(X\|F)$
Boja=Ljubičast	$\frac{1}{4}$	$\frac{1}{2}$
Boja=Žut	$\frac{3}{4}$	$\frac{1}{2}$
Veličina=Mali	$\frac{3}{4}$	$\frac{1}{4}$
Veličina=Veliki	$\frac{1}{4}$	$\frac{3}{4}$
Vrsta=Duguljast	$\frac{1}{2}$	$\frac{1}{2}$
Vrsta=Okrugao	$\frac{1}{2}$	$\frac{1}{2}$
Osoba=Odrasla	$\frac{1}{2}$	$\frac{1}{4}$
Osoba=Dete	$\frac{1}{2}$	$\frac{3}{4}$

$P(T) = \frac{1}{2}$
$P(F) = \frac{1}{2}$

Klasifikacija test instanci:

- $X_1 = (Boja = Ljubicast, Velicina = Mali, Vrsta = Okrugao, Osoba = Odrasla)$

  $P(T|X_1) = \frac{P(Boja=Ljubicast|T)*P(Velicina=Mali|T)*P(Vrsta=Okrugao|T)*P(Osoba=Odrasla|T)*P(T)}{P(X_1)} = \frac{\frac{1}{4}*\frac{3}{4}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}}{P(X_1)} = \frac{\frac{3}{2^7}}{P(X_1)}$

  $P(F|X_1) = \frac{P(Boja=Ljubicast|F)*P(Velicina=Mali|F)*P(Vrsta=Okrugao|F)*P(Osoba=Odrasla|F)*P(F)}{P(X_1)} = \frac{\frac{1}{2}*\frac{1}{4}*\frac{1}{2}*\frac{1}{4}*\frac{1}{2}}{P(X_1)} = \frac{\frac{1}{2^7}}{P(X_1)}$

  Kako je $P(T|X_1) > P(F|X_1)$, instancu X_1 klasifikujemo klasom T.

- $X_2 = (Boja = Zut, Velicina = Mali, Vrsta = Okrugao, Osoba = Odrasla)$

  $P(T|X_2) = \frac{P(Boja=Zut|T)*P(Velicina=Mali|T)*P(Vrsta=Okrugao|T)*P(Osoba=Odrasla|T)*P(T)}{P(X_2)} = \frac{\frac{3}{4}*\frac{3}{4}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}}{P(X_2)} = \frac{\frac{9}{2^7}}{P(X_2)}$

  $P(F|X_2) = \frac{P(Boja=Zut|F)*P(Velicina=Mali|F)*P(Vrsta=Okrugao|F)*P(Osoba=Odrasla|F)*P(F)}{P(X_2)} = \frac{\frac{1}{2}*\frac{1}{4}*\frac{1}{2}*\frac{1}{4}*\frac{1}{2}}{P(X_2)} = \frac{\frac{1}{2^7}}{P(X_2)}$

  Kako je $P(T|X_2) > P(F|X_2)$, instancu X_2 klasifikujemo klasom T.

- $X_3 = (Boja = Ljubicast, Velicina = Veliki, Vrsta = Okrugao, Osoba = Dete)$

  $P(T|X_3) = \frac{P(Boja=Ljubicast|T)*P(Velicina=Veliki|T)*P(Vrsta=Okrugao|T)*P(Osoba=Dete|T)*P(T)}{P(X_3)} = \frac{\frac{1}{4}*\frac{1}{4}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}}{P(X_3)} = \frac{\frac{1}{2^7}}{P(X_3)}$

$P(F|X_3) = \frac{P(Boja=Ljubicast|F)*P(Velicina=Veliki|F)*P(Vrsta=Okrugao|F)*P(Osoba=Dete|F)*P(F)}{P(X_3)} = \frac{\frac{1}{2}*\frac{3}{4}*\frac{1}{2}*\frac{3}{4}*\frac{1}{2}}{P(X_3)} = \frac{\frac{9}{2^7}}{P(X_3)}$

Kako je $P(T|X_3) < P(F|X_3)$, instancu X_3 klasifikujemo klasom F.

- $X_4 = (Boja = Ljubicast, Velicina = Veliki, Vrsta = Duguljast, Osoba = Odrasla)$

$P(T|X_4) = \frac{P(Boja=Ljubicast|T)*P(Velicina=Veliki|T)*P(Vrsta=Duguljast|T)*P(Osoba=Odrasla|T)*P(T)}{P(X_4)} = \frac{\frac{1}{4}*\frac{1}{4}*\frac{1}{2}*\frac{1}{2}*\frac{1}{2}}{P(X_4)} = \frac{\frac{1}{2^7}}{P(X_4)}$

$P(F|X_4) = \frac{P(Boja=Ljubicast|F)*P(Velicina=Veliki|F)*P(Vrsta=Duguljast|F)*P(Osoba=Odrasla|F)*P(F)}{P(X_4)} = \frac{\frac{1}{2}*\frac{3}{4}*\frac{1}{2}*\frac{1}{4}*\frac{1}{2}}{P(X_4)} = \frac{\frac{3}{2^7}}{P(X_4)}$

Kako je $P(T|X_4) < P(F|X_4)$, instancu X_4 klasifikujemo klasom F.

Preciznost izračunata na test instancama je $\frac{3}{4}$ jer su 3 instance (X_1, X_2 i X_4) od 4 dobro klasifikovane.

2 Klasifikacija teksta

2.1 Term-matrica i $tf - idf$ mera

Pri obradi tekstualnih dokumenata obično se primenjuju sledeći koraci:

1. eliminacija stop reči. Stop reči je skup reči nekog jezika koje se često upotrebljavaju, npr. veznici. Kako se one nalaze u svakom tekstu, obično nisu zanimljive pri analizi.
2. svođenje reči na koren
3. pravljenje term-matrice

U term-matrici atributi su termi (reči), a broj atributa je veličina rečnika. Jedan dokument predstavlja jednu instancu i podaci o njemu su predstavljeni u jednom redu. Za svaki dokument i svaki term (reč) čuva se broj pojavljivanja tog terma u tom dokumentu. Primer term matrice je dat u tabeli 1.

	term-matrica					
tekst	beijing	chinese	japan	macao	shanghai	tokyo
Chinese Beijing Chinese	1	2	0	0	0	0
Chinese Chinese Shanghai	0	2	0	0	1	0
Chinese Macao	0	1	0	1	0	0
Tokyo Japan Chinese	0	1	1	0	0	1

Tabela 1: Primer term-matrice sa brojem pojavljivanja terma u tekstu

Umesto broja pojavljivanja terma može da se koristi $tf-idf$ (*term-frequency - inverse document frequency*) mera u kojoj je

- *tf* - frekvencija reči (*term-frequency*)
- *idf* - inverzna frekvencija dokumenta (*inverse document frequency*) je težina kojom se određuje značajnost terma u kolekciji tekstualnih dokumenata

Ako su:

- t - term
- d - dokument
- n - ukupan broj dokumenata
- *df(t)* - broj dokumenata koji sadrže term t

formule za $tf - idf$ meru i *idf* su:

$$tf - idf(t, d) = tf(t, d) * idf(t)$$

$$idf(t) = \log[n/df(t)] + 1^1$$

Mera $tf - idf$ smanjuje uticaj terma koji se često javlja u datom korpusu, a zbog 1 u *idf(t)* term koji se javlja u svim dokumentima neće u potpunosti biti ignosrisan.

2.2 Naivni Bajes za klasifikaciju teksta

Za klasifikaciju teksta koristiti se varijanta naivnog Bajesa - multinomijalni naivni Bajes. Na osnovu trening skupa, svakoj klasi C se dodeljuje vektor parametara $\Theta_c = (\Theta_{c1}, \Theta_{c2}, ..., \Theta_{cn})$, gde je n broj terma (atributa), a Θ_{ci} verovatnoća da se term i pojavi u instanci koja pripada klasi C. Verovatnoća Θ_{ci} se računa prema formuli

$$\Theta_{ci} = \frac{N_{ci} + \alpha}{N_c + \alpha * n}$$

gde je

- N_{ci} broj pojavljivanja terma (reči) i u dokumentima klase C
- N_c ukupan broj pojavljivanja svih reči u klasi C
- α parametar za uglađivanje koji se zadaje i služi za za određivanje verovatnoće za vrednosti koje se ne pojavljuju u trening skupu kako se ne bi pojavila verovatnoća 0 pri računu.

Klasifikacija test dokumenta d sa termima $\langle t_1, t_2, ...t_{nd} \rangle$ se vrši računanjem

$$\widehat{C} = \arg\max_C P(C) \prod_{i=1}^{nd} P(t_i|C) = \arg\max_C P(C) \prod_{i=1}^{nd} \Theta_{ci}$$

Radi lakšeg izračunavanja može se koristiti

$$\widehat{C} = \arg\max_C [\log P(C) + \sum_{i=1}^{nd} \log P(t_i|C)]$$

[1] U literaturi se mogu naći i druge formule za *idf*

2.3 Zadaci

3. Dati su podaci :

Id teksta	reči u dokumentu	klasa
1	Chinese Beijing Chinese	yes
2	Chinese Chinese Shanghai	yes
3	Chinese Macao	yes
4	Tokyo Japan Chinese	no

Primenom naivnog Bajesa za klasifikaciju teksta klasifikovati tekst X=*Chinese Chinese Chinese Tokyo Japan* ako je $\alpha = 1$.

Rešenje

Verovatnoće za klase u trening skupu su:

$P(yes) = \frac{3}{4}$
$P(no) = \frac{1}{4}$

Verovatnoće za reči u klasi *yes* (Θ_{ci}, videti definiciju na strani 6) u trening skupu:

$P(Chinese|yes) = \frac{5+1}{8+6} = \frac{6}{14} = \frac{3}{7}$
Vrednosti u razlomku su:

- 5 je ukupan broj pojavljivanja reči *Chinese* u tekstovima klase *yes*
- 8 je ukupan broj reči u tekstovima klase *yes*
- 6 je broj različitih reči koji se javlja u celom skupu

$P(Tokyo|yes) = \frac{0+1}{8+6} = \frac{1}{14}$
$P(Japan|yes) = \frac{0+1}{8+6} = \frac{1}{14}$

$P(yes|X) = P(yes) * P(Chinese|yes)^3 * P(Tokyo|yes) * P(Japan|yes) = \frac{3}{4} * \frac{3}{7}^3 * \frac{1}{14} * \frac{1}{14} \approx 0,0003$
Pri računanju $P(yes|X)$ je navedeno $P(Chinese|yes)^3$, jer se reč *Chinese* pojavljuje triputa u test instanci.

Verovatnoće za reči u klasi *no* su:

$P(Chinese|no) = \frac{1+1}{3+6} = \frac{2}{9}$
$P(Tokyo|no) = \frac{1+1}{3+6} = \frac{2}{9}$
$P(Japan|no) = \frac{1+1}{3+6} = \frac{2}{9}$

$P(no|X) = P(no) * P(Chinese|no)^3 * P(Tokyo|no) * P(Japan|no) = \frac{1}{4} * \frac{2}{9}^3 * \frac{2}{9} * \frac{2}{9} \approx 0,0001$

Kako je $P(yes|X) > P(no|X)$ test instanci dodeljujemo klasu *yes*.

3 Naivni Bajesovski klasifikatori u biblioteci scikit-learn

3.1 Klasifikacija teksta u biblioteci scikit-learn

3.1.1 Izdvajanje terma iz teksta

Pri obradi tekstualnih dokumenata možemo da koristimo klase modula *sklearn.feature_extraction.text*:

- **CountVectorizer** za pretvaranje kolekcije tekst dokumenata u term-matricu sa brojem pojavljivanja terma u dokumentu
- **TfidfVectorizer** za pretvaranje kolekcije tekst dokumenata u term-matricu u kojoj atributi sadrže vrednosti dobijene primenom *tf-idf* mere
- **TfidfTransformer** za pretvaranje term-matrice sa brojem pojavljivanja u matricu sa *tf-idf* atributima

Neki od parametara koji su zajednički za klase **TfidfVectorizer** i **CountVectorizer** su:

- *input* - šta je ulaz ('filename', 'file', 'content') (default='content')
- *lowercase* - sva slova će biti pretvorena u mala pre obrade (default=True)
- *stop_words* - reči koje će biti uklonjene (default='english')
- *max_df* - ignoriše reči koje imaju dokument-frekvenciju iznad zadatog praga (zadaje se procenat ili broj dokumenata) (default=1.0)
- *min_df* - ignoriše reči koje imaju dokument-frekvenciju ispod zadatog praga (zadaje se procenat ili broj dokumenata) (default=1.0)
- *binary* - pravljenje binarne term-matrice. Term koji se javlja u dokumentu ima vrednost 1 umesto broja pojavljivanja. (default=False)

Neki od parametara koji su zajednički za klase **TfidfVectorizer** i **TfidfTransformer** su:

- *norm* - normalizacija vrednosti jedne instance: l1, l2 ili None (default='l2')
  - l2 - zbir kvadrata vrednosti atributa za jednu instancu je 1. Proizvod dve instance je njihova kosinusna sličnost, a **kosinusna sličnost je mera koja se često koristi u analizi tekstova.**
  - l1 - zbir vrednosti atributa za jednu instancu je 1
- *use_idf* - da li da se koriste težine kojima se određuje značajnost termova u kolekciji tekstualnih dokumenata. (default=True)

Neke metode navedenih klasa za obradu teksta:

- *fit* - uči rečnik na osnovu zadatog skupa
- *fit_transform* - uči rečnik i vraća term-matricu na osnovu zadatog skupa
- *transform* - pretvara zadate tekstove u term-matricu
- *get_feature_names* - vraća imena atributa napravljene term-matrice (za klase *TfidfVectorizer* i *CountVectorizer*)
- *get_stop_words* - vraća stop reči (za klase *TfidfVectorizer* i *CountVectorizer*)

3.1.2 Izdvajanje terma iz rečnika

Primenom klase *sklearn.feature_extraction.DictVectorizer* lista rečnika sa podacima u obliku atribut-vrednost može da se transformiše u term-matricu. Jedan rečnik predstavlja jednu instancu.

- metode
  - *fit* - uči rečnik na osnovu zadatog skupa
  - *fit_transform* - uči rečnik i vraća term-matricu
  - *get_feature_names* - vraća imena atributa
  - transform - pretvara zadati rečnik(e) u term-matricu

3.1.3 Naivni Bajesovski klasifikator za klasifikaciju teksta

Algoritam multinomijalni naivni Bajes je implementiran u klasi *sklearn.naive_bayes.MultinomialNB*. Karakteristike klase su:

- parametri
  - *alpha* - parametar za uglađivanje (default=1.0)
  - *fit_prior* - da li se verovatnoće klasa uče iz trening skupa (default=True)
  - *class_prior* - zadaju se verovatnoće klasa (default=None)
- atributi
  - *class_count_* - izračunati broj instanci po klasama tokom pravljenja modela
  - *feature_count_* - izračunati broj instanci za svaku klasu i svaki term tokom pravljenja modela
- metode
  - *fit* - pravi model na osnovu zadatog skupa
  - *predict* - određuje klase test instancama
  - *predict_proba* - vraća procenjenu verovatnoću pripadnosti svakoj od klasa za test instance

Primeri u programskom jeziku Python:

1. Klasifikacija test instance korišćenjem trening skupa iz 3. zadatka - **zad3_python.py**
2. Dat je skup sa podacima iz novinskih članaka - ebart. Članci su podeljeni prema klasi kojoj pripadaju u direktorijume: Ekonomija, HronikaKriminal, KulturaZabava, Politika i Sport. Svaki članak je obrađen: uklonjene su stop reči i svaka reč je zamenjena svojim korenom, a zatim je izvršeno prebrojavanje reči. Rezultat obrade svakog članka je sačuvan u zasebnoj datoteci. U dobijenoj datoteci koja odgovara jednom članku, u jednom redu su podaci o jednom korenu reči - koren reči i broj pojavljivanja tog korena u tom članku.

   Primeniti klasifikaciju nad ovim skupom primenom različitih algoritama za klasifikaciju i na standardni izlaz ispisati izveštaj o uspešnosti za svaki od napravljenih modela.

   ebart.py

3.2 Naivni Bajesovski klasifikator za neprekidne atribute

Za klasifikaciju skupa sa neprekidnim atributima može da se koristi Gausov naivni Bajesovski algoritam, koji je implementiran u klasi **sklearn.naive_bayes.GaussianNB**. Uslovna vreovarnoća pojavljivanja x_i u datoj klasi C se računa po formuli:

$$P(x_i|C) = \frac{1}{\sqrt{2\pi\sigma_c^2}} e(-\frac{(x_i - \mu_c)^2}{2\sigma_c^2})$$

Podaci o klasi:

- parametar:
  - *priors* - verovatnoće klasa (default=None). Ako se ne zada vrednost paramtera, verovatnoće klasa se računaju na osnovu trening skupa.
- atributi
  - *class_count_* - izračunat broj instanci po klasama u trening skupu
  - *class_prior_* - verovatnoća za svaku od klasa
  - *theta_* - srednja vrednost atributa po klasi
  - *sigma_* - varijansa atributa po klasi
- metode
  - *fit* -pravli model na osnovu zadatog skupa
  - *predict* - određuje klase test instancama
  - *predict_proba* - vraća procenjenu verovatnoću pripadnosti svakoj od klasa za test instance

Primer:

3. Klasifikacija skupa o perunikama primenom algoritma naivni Bajes - **iris_nb.py**