Funkcijų aproksimavimas:
Aproksimavimas daugianariais

Funkcijų aproksimavimas. Uždavinio formuluotė

Aproksimavimas – tai funkcijos, kurios reikšmės aproksimavimo taškuose būtų *kiek galima mažiau nutolusios* nuo duotųjų reikšmių, radimas.

1. **Kreivė $y=f(x)$ <u>neprivalo</u> praeiti per duotus taškus (x_i,y_i), t. y. $f(x_i)=y_i$, $i=1, 2, ..., n$;**
2. **funkcijos $f(x)$ analitinė išraiška neturi būti labai sudėtinga;**
3. **funkcija $f(x)$ turi būti nesunkiai integruojama ir diferencijuojama;**
4. **funkcija $f(x)$ turi būti nesunkiai surandama (pvz., jos parametrai apskaičiuojami pagal žinomas formules, arba sprendžiant tiesinių lygčių sistemą).**

Funkcijų aproksimavimas *mažiausių kvadratų* metodu

Taškų seka, kurią reikia aproksimuoti: $(x_i, y_i)\,,\quad i = 1,\ldots,n$

$$\Psi = \frac{1}{2}\sum_{j=1}^{n}\left(f\left(x_j\right) - y_j \right)^2$$

Aproksimacijos kokybės įverčio funkcija

$$\min_{f} \Psi$$

ieškoma funkcija

Funkcijų *interpoliavimas* yra atskiras funkcijų *aproksimavimo* uždavinio atvejis

Aproksimavimas daugianariais *vienanarių bazėje*

Taškų seka, kurią reikia aproksimuoti: $\left(x_i, y_i\right), \quad i = 1, \dots, n$

Duotų taškų skaičius ir aproksimuojančių funkcijų skaičius gali nesutapti: $m \le n$

Aproksimuojančios funkcijos ieškosime parinktų *bazinių funkcijų* tiesinės kombinacijos pavidale

$$f(x) = \begin{bmatrix} g_1(x) & g_2(x) & \dots & g_{m-1}(x) & g_m(x) \end{bmatrix} \begin{Bmatrix} c_1 \\ c_2 \\ \vdots \\ c_{m-1} \\ c_m \end{Bmatrix} = \left[\mathbf{g}(x)\right]\{\mathbf{c}\}$$

Bazinės funkcijos

koeficientai

Aproksimavimo kokybės įverčio funkcija užrašoma matricomis

$$f\left(x_j\right)-y_j$$

$$\begin{Bmatrix} f(x_1)-y_1 \\ f(x_2)-y_2 \\ \vdots \\ f(x_n)-y_n \end{Bmatrix} = \begin{bmatrix} g_1(x_1) & g_2(x_1) & \cdots & g_m(x_1) \\ g_1(x_2) & g_2(x_2) & \cdots & g_m(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ g_1(x_n) & g_2(x_n) & \cdots & g_m(x_n) \end{bmatrix} \begin{Bmatrix} c_1 \\ c_2 \\ \vdots \\ c_m \end{Bmatrix} - \begin{Bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{Bmatrix} = [\mathbf{G}]_{n\times m}\{\mathbf{c}\}_{m\times 1} - \{\mathbf{y}\}_{n\times 1}$$

$$\min_{\mathbf{c}} \Psi = \frac{1}{2}\left(\mathbf{Gc}-\mathbf{y}\right)^T\left(\mathbf{Gc}-\mathbf{y}\right)$$

ieškomi koeficientai

$$\left(\frac{\partial \Psi}{\partial \mathbf{c}}\right)^T = 0$$

Pagal priimtus žymėjimus, skaliarinės funkcijos išvestinė pagal vektorinį argumentą (gradientas) yra matrica-eilutė. Standartinio pavidalo lygčių sistemą gausime, jį pateikę stulpeliu, t.y. transponuodami

Pagalbinė medžiaga: matricų ir vektorių išraiškų diferencijavimo formulės

$$f = \mathbf{x}^T \mathbf{x} = \sum_{i=1}^{n} x_i^2;$$

$$\left[\frac{\partial f}{\partial \mathbf{x}}\right] = \left\{\frac{\partial f}{\partial x_1} \quad \frac{\partial f}{\partial x_2} \quad \cdots \quad \frac{\partial f}{\partial x_n}\right\} = 2\left\{x_1 \quad x_2 \quad \cdots \quad x_n\right\} = 2\mathbf{x}^T; \quad \longleftarrow \quad \text{gradientas}$$

gradientas

$$f = \mathbf{x}^T \mathbf{A}\mathbf{x} = \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij} x_i x_j;$$

$$\left[\frac{\partial f}{\partial \mathbf{x}}\right] = \left\{\frac{\partial f}{\partial x_1} \quad \frac{\partial f}{\partial x_2} \quad \cdots \quad \frac{\partial f}{\partial x_n}\right\} = \left[\sum_{j=1}^{n}\left(a_{1j} + a_{j1}\right)x_j \quad \sum_{j=1}^{n}\left(a_{2j} + a_{j2}\right)x_j \quad \cdots \quad \sum_{j=1}^{n}\left(a_{nj} + a_{jn}\right)x_j\right] = \mathbf{x}^T\left(\mathbf{A} + \mathbf{A}^T\right)$$

$$\mathbf{f} = \mathbf{A}\mathbf{x}$$

$$\left[\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right] = \left[\frac{\partial\left(\mathbf{A}\mathbf{x}\right)}{\partial \mathbf{x}}\right] = \mathbf{A}\left[\frac{\partial \mathbf{x}}{\partial \mathbf{x}}\right] = \mathbf{A}\begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} = \mathbf{A}; \quad \longleftarrow \quad \text{Jakobio matrica}$$

Aproksimavimo kokybės įverčio minimumo sąlyga

$$\Psi = \frac{1}{2}\left(\mathbf{Gc}-\mathbf{y}\right)^T\left(\mathbf{Gc}-\mathbf{y}\right)$$

$$\frac{\partial\left(\mathbf{x}^T\mathbf{x}\right)}{\partial \mathbf{x}} = 2\mathbf{x}^T$$

$$\frac{\partial\left(\mathbf{Ax}\right)}{\partial \mathbf{x}} = \mathbf{A}$$

$$\frac{\partial\Psi}{\partial\mathbf{c}} = \frac{\partial\Psi}{\partial\left(\mathbf{Gc}-\mathbf{y}\right)}\frac{\partial\left(\mathbf{Gc}-\mathbf{y}\right)}{\partial\mathbf{c}} = \left(\mathbf{Gc}-\mathbf{y}\right)^T\mathbf{G}$$

Pagal priimtus žymėjimus, skaliarinės funkcijos išvestinė pagal vektorinį argumentą yra matrica-eilutė

$$\left(\frac{\partial\Psi}{\partial\mathbf{c}}\right)^T = \mathbf{G}^T\left(\mathbf{Gc}-\mathbf{y}\right) = 0$$

Gautą priklausomybę laikysime lygčių sistema, todėl standartinis jos pavidalas gaunamas transponuojant

$$\left(\left(\mathbf{G}^T\right)_{m\times n}\mathbf{G}_{n\times m}\right)_{m\times m}\mathbf{c}_{m\times 1} = \left(\mathbf{G}^T\right)_{m\times n}\mathbf{y}_{n\times 1}$$

Lygčių skaičius toks, kiek yra bazinių funkcijų, t.y. *m*.
Jis nepriklauso nuo duotų aproksimavimo taškų skaičiaus

Koeficientų apskaičiavimas aproksimavimo ir interpoliavimo atveju

aproksimavimas

$$\left(\left(\mathbf{G}^T\right)_{m\times n}\mathbf{G}_{n\times m}\right)_{m\times m}\mathbf{c}_{m\times 1}=\left(\mathbf{G}^T\right)_{m\times n}\mathbf{y}_{n\times 1}$$

Kairiojo matricinio daugiklio negalima išbraukti, kadangi pasikeistų lygčių skaičius

$$m=n$$

$$\cancel{\left(\mathbf{G}^T\right)_{n\times n}}\mathbf{G}_{n\times n}\mathbf{c}_{n\times 1}=\cancel{\left(\mathbf{G}^T\right)_{n\times n}}\mathbf{y}_{n\times 1}$$

Kairįjį daugiklį atmetus, lygčių skaičius nepakinta, o sistema išlieka išsprendžiama

$$\mathbf{G}_{n\times n}\mathbf{c}_{n\times 1}=\mathbf{y}_{n\times 1}$$

interpoliavimas

Aproksimavimo vienanariais eiga

```
function G=base(m,x)
  for i=1:m, G(:,i)=x.^(i-1); end
return,end
```

`G=base(m,X(1:n))`

Duoti aproksimavimo taškai X

$$\mathbf{G}=\begin{bmatrix} g_1(x_1) & g_2(x_1) & \cdots & g_m(x_1) \\ g_1(x_2) & g_2(x_2) & \cdots & g_m(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ g_1(x_n) & g_2(x_n) & \cdots & g_m(x_n) \end{bmatrix}=\begin{bmatrix} 1 & x_1 & \cdots & x_1^{m-1} \\ 1 & x_2 & \cdots & x_2^{m-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \cdots & x_n^{m-1} \end{bmatrix}$$

$$\left(\mathbf{G}^T\mathbf{G}\right)\mathbf{c}=\mathbf{G}^T\mathbf{y}$$

Aproksimuojančios funkcijos vaizdavimas:

$$f(x)=\left[\mathbf{g}(x)\right]\left\{\mathbf{c}\right\}$$

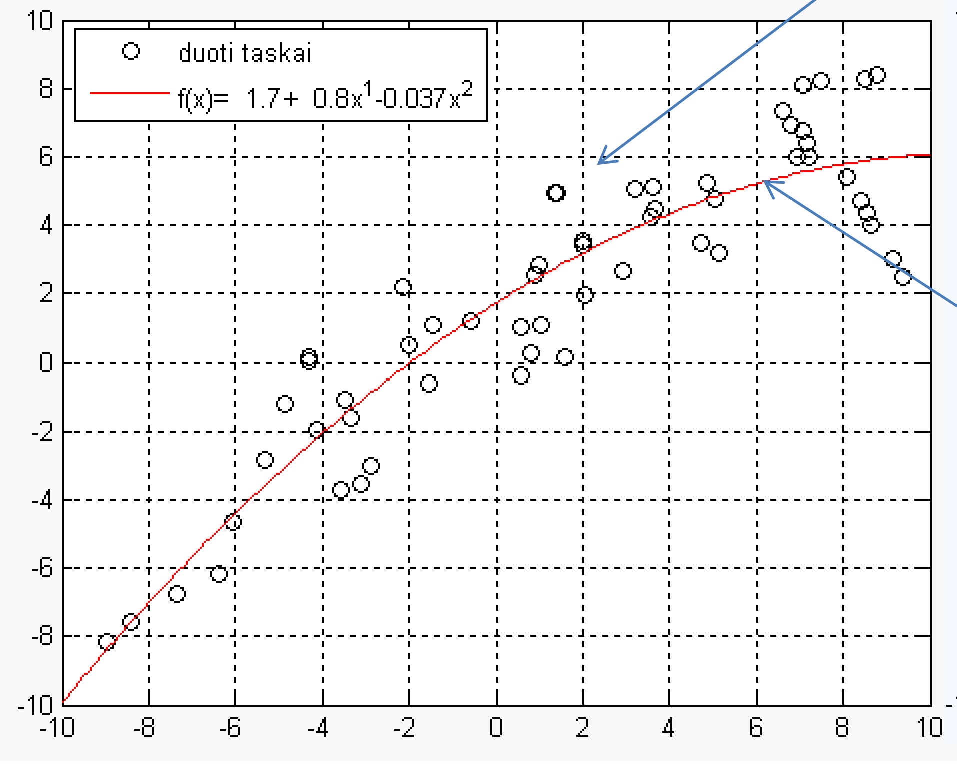
Vaizdavimo taškai (linspace)

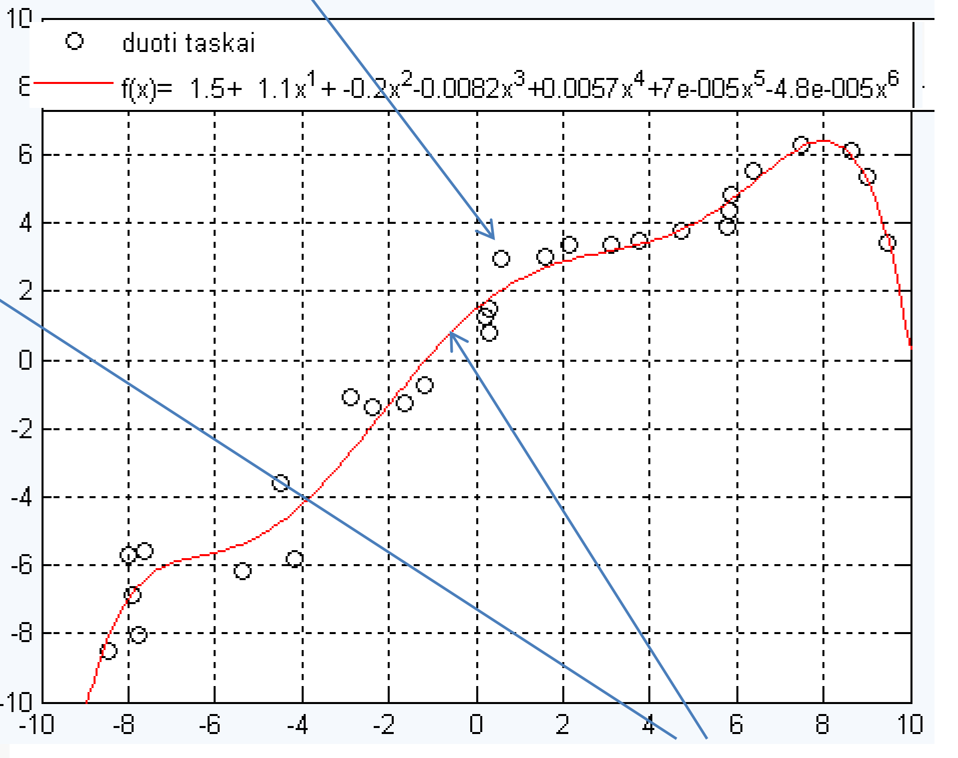
Aproksimavimas daugianariais *vienanarių bazėje*

Pvz_SMA_9_1_aproksimavimas_vienanariu_baze

Taškų seka, kurią reikia aproksimuoti: $\left(x_i, y_i\right), \quad i = 1, \ldots, n$

Bazinės funkcijos: $\begin{bmatrix} 1 & x & x^2 & \ldots & x^{m-1} \end{bmatrix}$

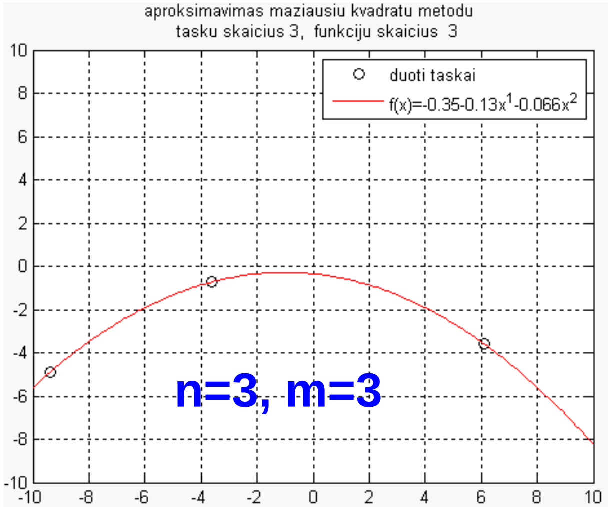


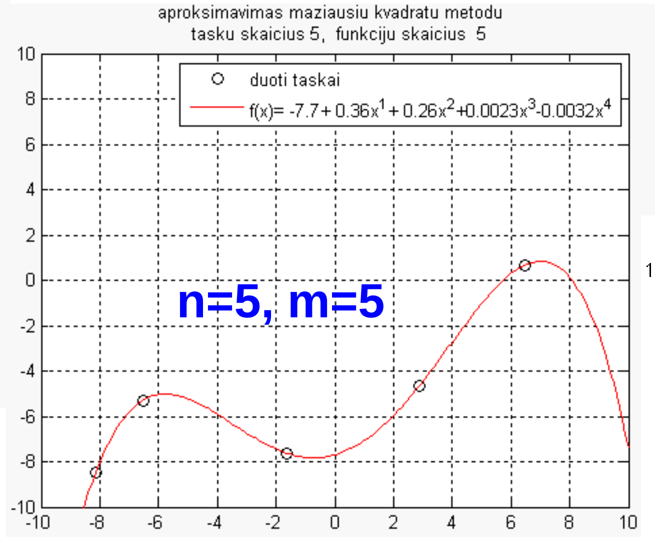


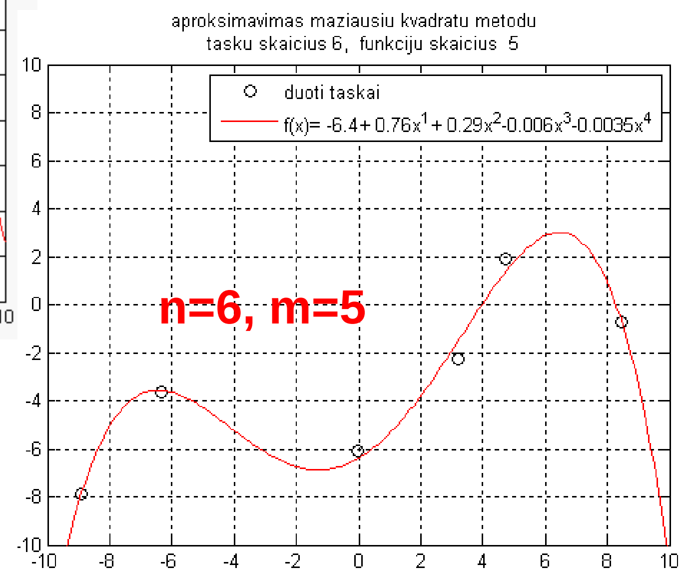
$$\mathbf{G}^T\mathbf{G}\mathbf{c} = \mathbf{G}^T\mathbf{y} \Rightarrow \mathbf{c} \Rightarrow \tilde{\mathbf{G}}\mathbf{c}$$

Kai *bazinių funkcijų skaičius* ir *aproksimuojamų taškų skaičius* sutampa (n=m), aproksimuojanti kreivė praeina per visus taškus:

$$\min_f \Psi = 0$$







Interpoliavimo uždavinys yra aproksimavimo uždavinio atskiras atvejis, kai *n*=*m*

- **Mažiausių kvadratų būdu sukurta aproksimuojanti kreivė dar vadinama *regresijos kreive***

  Regresijos **terminas yra naudojamas kaip antonimas žodžiui** ***progresuoti (tobulėti, išsiskirti iš vidurkio). Regresuoti – prarasti gerąsias savybes, artėti prie vidurkio. Regresijos kreivė aprašo taškų vidurines padėtis;***

- **Aproksimavimui apibūdinti dar naudojamas terminas *kreivės pritaikymas (curve fitting)***

Vektorinio argumento funkcijų aproksimavimas

Taškų aibė, kurią reikia aproksimuoti: $\left(x_i, y_i, z_i\right), \quad i = 1, \dots, n$

$$\Psi = \frac{1}{2}\sum_{j=1}^{n}\left(f\left(\underbrace{x_j, y_j}\right) - z_j \right)^2$$

Aproksimacijos kokybės įverčio funkcija

argumento vektorius

$$\min_f \Psi$$

ieškoma funkcija

$$f(x,y) = \left[g_1(x,y) \quad g_2(x,y) \quad \dots\right]\begin{Bmatrix} c_1 \\ c_2 \\ \vdots \end{Bmatrix} = \left[\mathbf{g}(x,y)\right]\{\mathbf{c}\}$$

Bazinių funkcijų vektorius

koeficientai

Vienanarių bazinių funkcijų matrica vektorinio argumento atveju

$$\mathbf{g}(x,y)=\begin{bmatrix}1 & x & \cdots & x^{m-1} & y & \cdots & y^{m-1}\end{bmatrix}$$

$$\mathbf{G}=\begin{bmatrix}1 & x_1 & \cdots & x_1^{m-1} & y_1 & \cdots & y_1^{m-1}\\ 1 & x_2 & \cdots & x_2^{m-1} & y_2 & \cdots & y_2^{m-1}\\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots\\ 1 & x_n & \cdots & x_n^{m-1} & y_n & \cdots & y_n^{m-1}\end{bmatrix}$$

$$\mathbf{G}^T\mathbf{G}\mathbf{c}=\mathbf{G}^T\mathbf{z} \quad \Rightarrow \quad \mathbf{c}$$

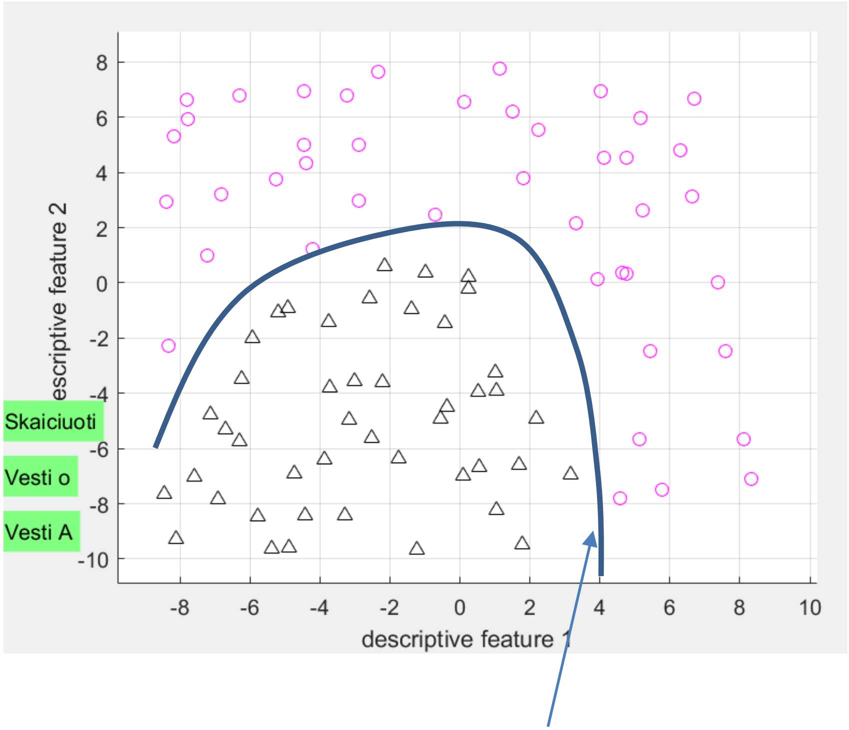
$$\mathbf{f}(x,y)=\begin{bmatrix}1 & x & \cdots & x^{m-1} & y & \cdots & y^{m-1}\end{bmatrix}\mathbf{c}$$

- **Parodėme, kad vektorinio argumento aproksimuojanti funkcija randama panašiai, kaip ir vieno argumento aproksimuojanti funkcija;**
- **Vieno argumento aproksimuojanti funkcija yra *regresijos kreivė f(x)***
- **Vektorinio argumento atveju aproksimuojanti funkcija yra *regresijos paviršius $f(x,y)$, kai kintamieji du) arba „hiperpaviršius“ $f(x_1,x_2,\ldots)$, kai kintamųjų daugiau nei du;***
- **Taikydami vektorinio argumento taškų aibės aproksimavimą galime išspręsti *klasifikavimo uždavinį, kai apytiksliai randama kreivė, plokštumoje atskirianti vieną nuo kitos dvi skirtingai pažymėtas taškų aibės dalis***

Klasifikavimo uždavinio sprendimas (1)

pavyzdys Pvz_SMA_9_01x_klasifikavimas_mouse.m

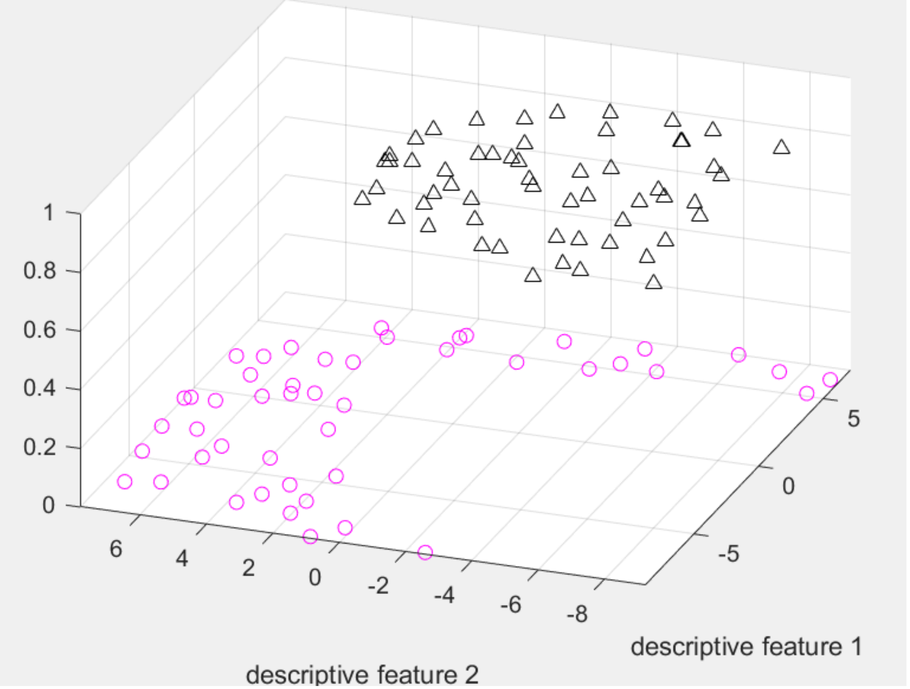
- **Plokštumoje duota taškų aibė, kuri susideda iš pagal dviejų *aprašančiųjų požymių (descriptive features)* reikšmes atskirtų klasių (pavyzdžiui, skritulių ir trikampių);**
- **Čia aprašantieji požymiai yra taškų x ir y koordintės;**



- **Reikia apytiksliai nustatyti klases atskiriančią liniją**

Klasifikavimo uždavinio sprendimas (2)

- **Išplečiame duotųjų taškų (t.y. *apmokančios aibės, training set*) vaizdavimą į 3D erdvę;**
- **Skirtingoms klasėms priklausantiems taškams suteikiame z koordinates 0 ir 1 atitinkamai:**



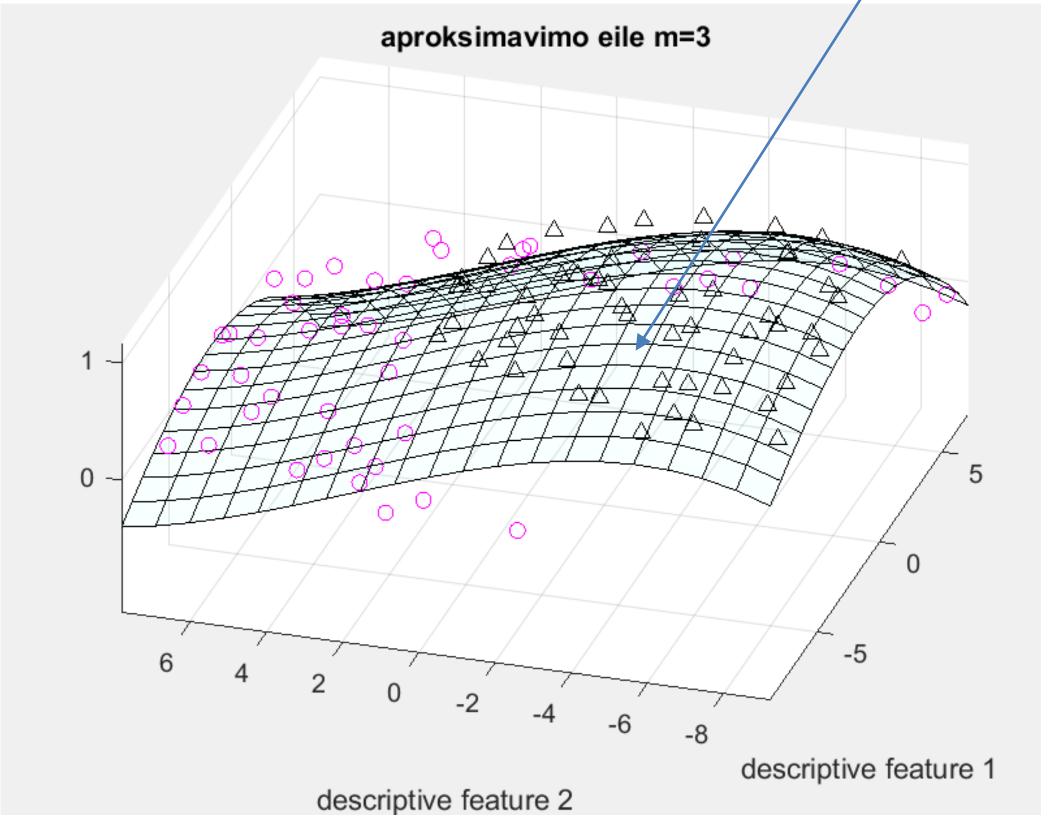
Klasifikavimo uždavinio sprendimas (3)

- **Parinkę aproksimavimo eilę (pavyzdžiui, m-1=3) randame aproksimuojantį paviršių:**

$$\mathbf{g}(x,y)=\begin{bmatrix}1 & x & x^2 & x^3 & y & y^2 & y^3\end{bmatrix}$$

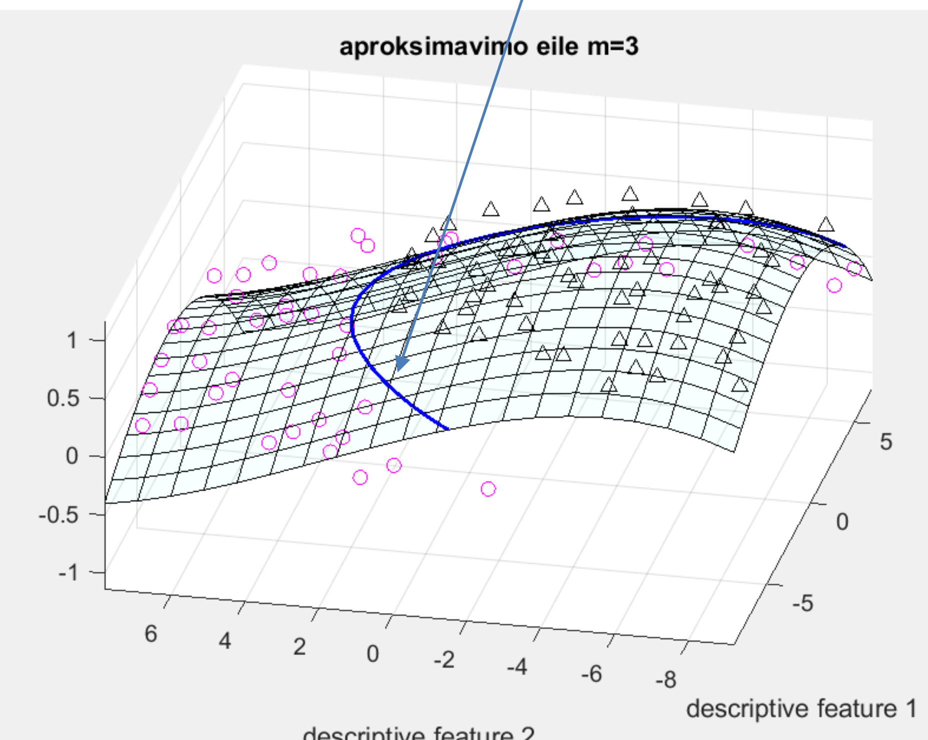
$$\mathbf{G}^T\mathbf{G}\mathbf{c}=\mathbf{G}^T\mathbf{z}$$

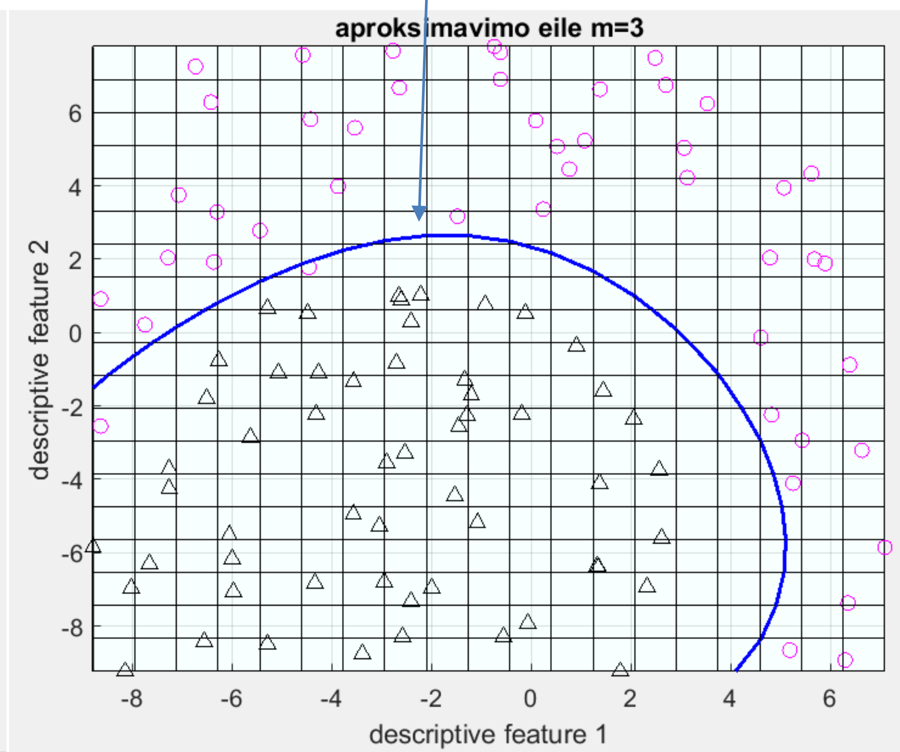
$$\mathbf{f}(x,y)=\begin{bmatrix}1 & x & x^2 & x^3 & y & y^2 & y^3\end{bmatrix}\mathbf{c}$$



Klasifikavimo uždavinio sprendimas (4)

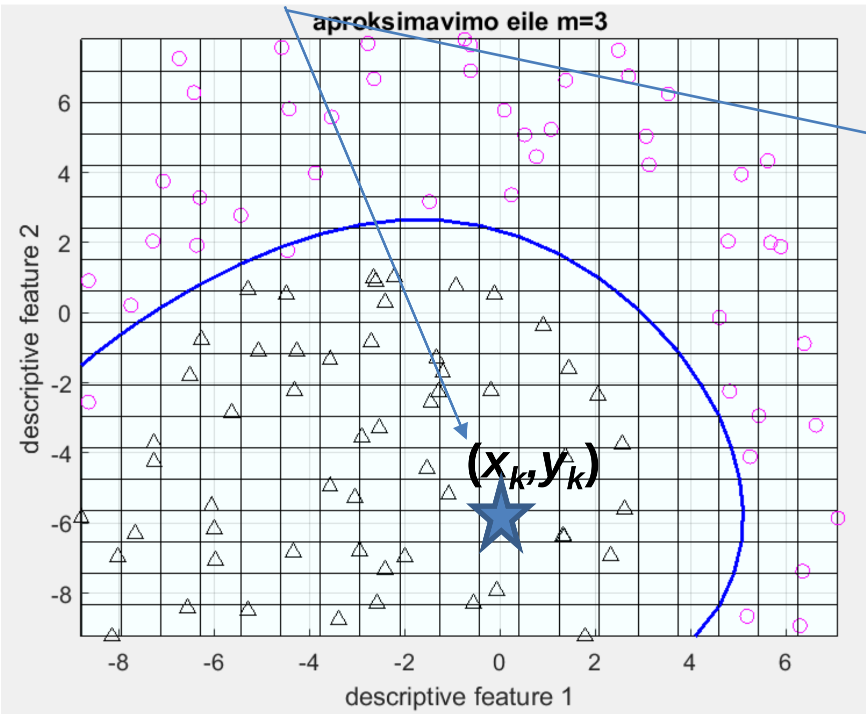
- **Taškų klases skiria linija, kuria gautas paviršius kerta plokštumą *z*=0.5 (kadangi klasėms buvome parinkę reikšmes z=0 ir z=1);**
- **Rezultatą patogiau interpretuoti nagrinėjant projekciją, statmeną ašiai Oz**

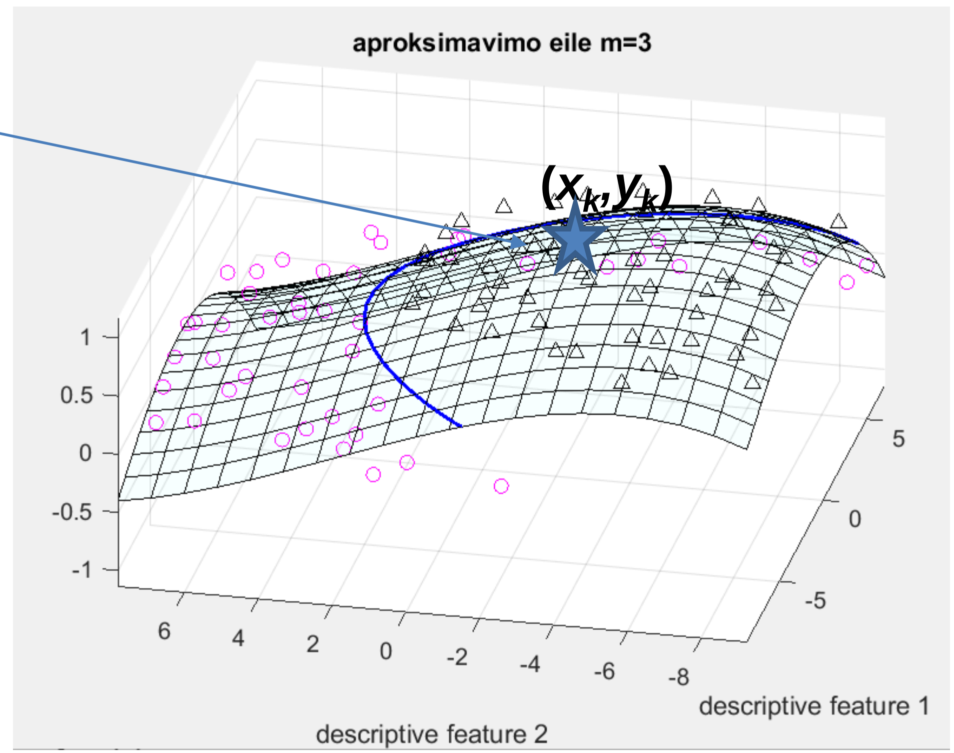




Klasifikavimo uždavinio sprendimas (5)

- **Naujai duotas taškas (x_k,y_k) priklauso trikampiams, jeigu f(x_k,y_k)>=0.5 . Priešingu atveju jis priklausytų skrituliams;**





- **Analogiškai skaičiuotume ir esant didesniam aprašančiųjų požymių skaičiui, tačiau rezultatą grafiškai pavaizduoti būtų keblu**

SMA_09-1_Klausimai savikontrolei:

1. Suformuluokite funkcijos aproksimavimo uždavinį;
2. Naudodamiesi literatūra paaiškinkite, kaip aproksimavimo uždavinys sprendžiamas mažiausių kvadratų metodu;
3. Koks yra ryšys tarp aproksimuojamų taškų ir bazinių funkcijų skaičiaus;
4. Kokiu atveju aproksimavimo ir interpoliavimo uždaviniai yra tapatūs;
5. Kokia problema iškyla, jeigu bazinių funkcijų parenkame daugiau, nei yra aproksimuojamų taškų;