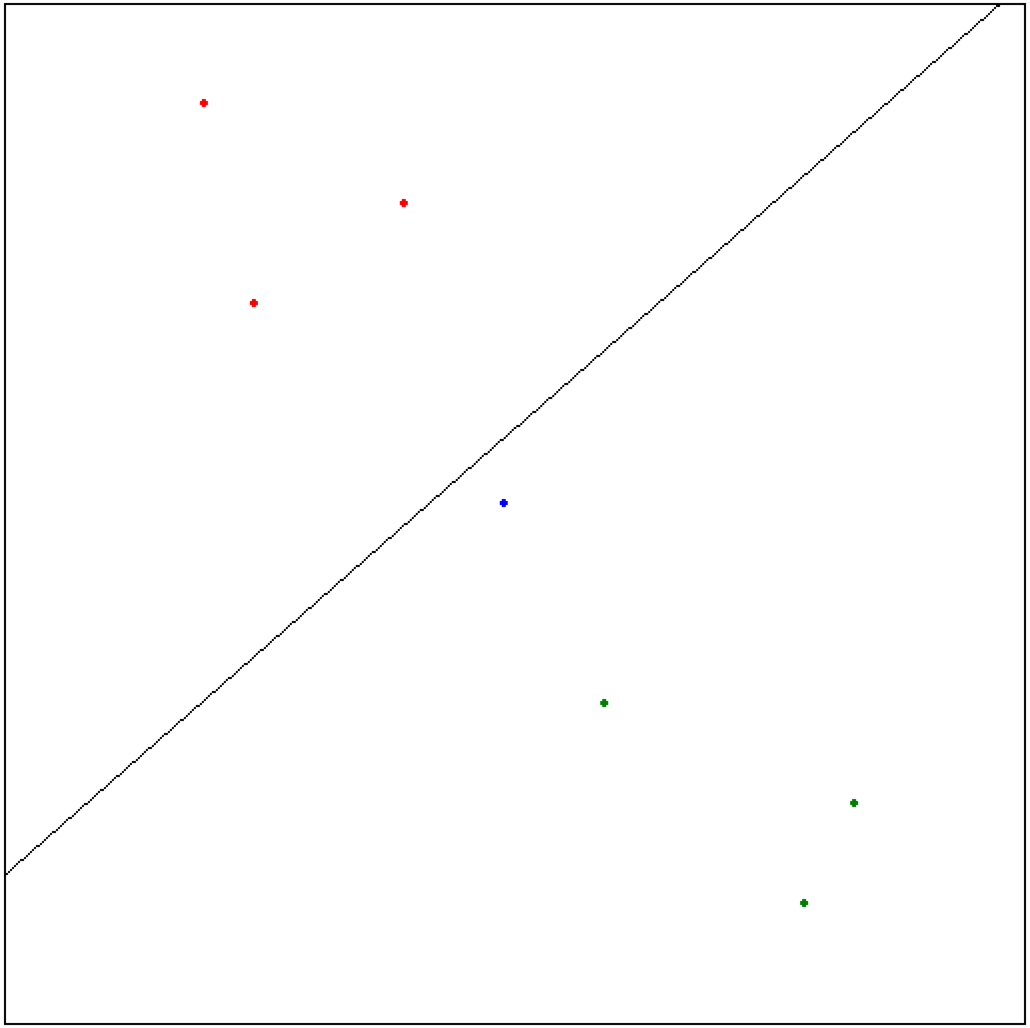
Optimālā atdalošā hiperplakne

Uzdevumu nostādne

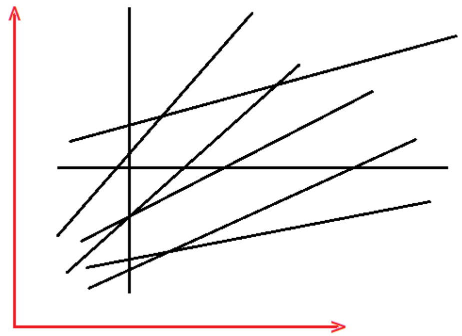
- Uzdevums: atrast optimālo atdalošo hiperplakni (hipervektoru)

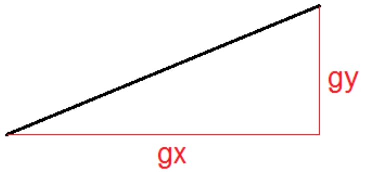


Optimālā atdalošā hiperplakne

- Ģenerējam nejaušo vektoru kopu (vektoru leņķi atrodas robežās no 90 līdz 270 grādiem). No tiem pēc tam atradīsim optimālo.
- Vektorus aprakstam ar gx, gy (atceramies trigonometriju un Pitagora teorēmu?)
- Programmā tas izskatās šādi:

```
Randomize;
for i:=0 to n do
begin
   lenkis:=random(180)+90
   gx:=cos(lenkis*pi/180);
   gy:=sin(lenkis*pi/180);
end;
```

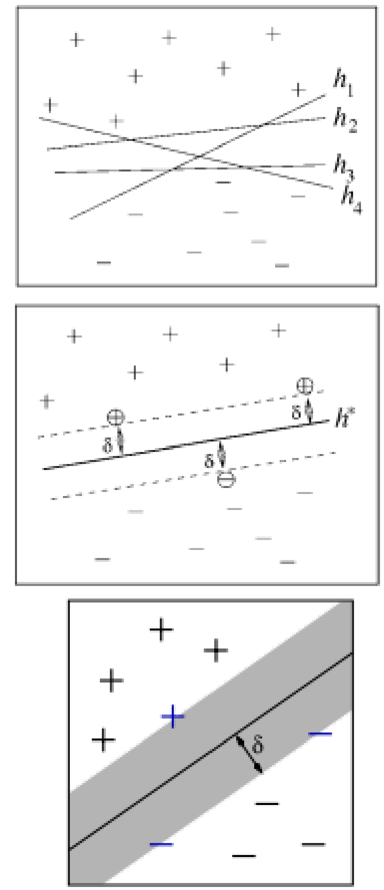




Vektora numurs	gx	gy
1	-0.99	0.10
...	...	...
n	-0.13	0.99

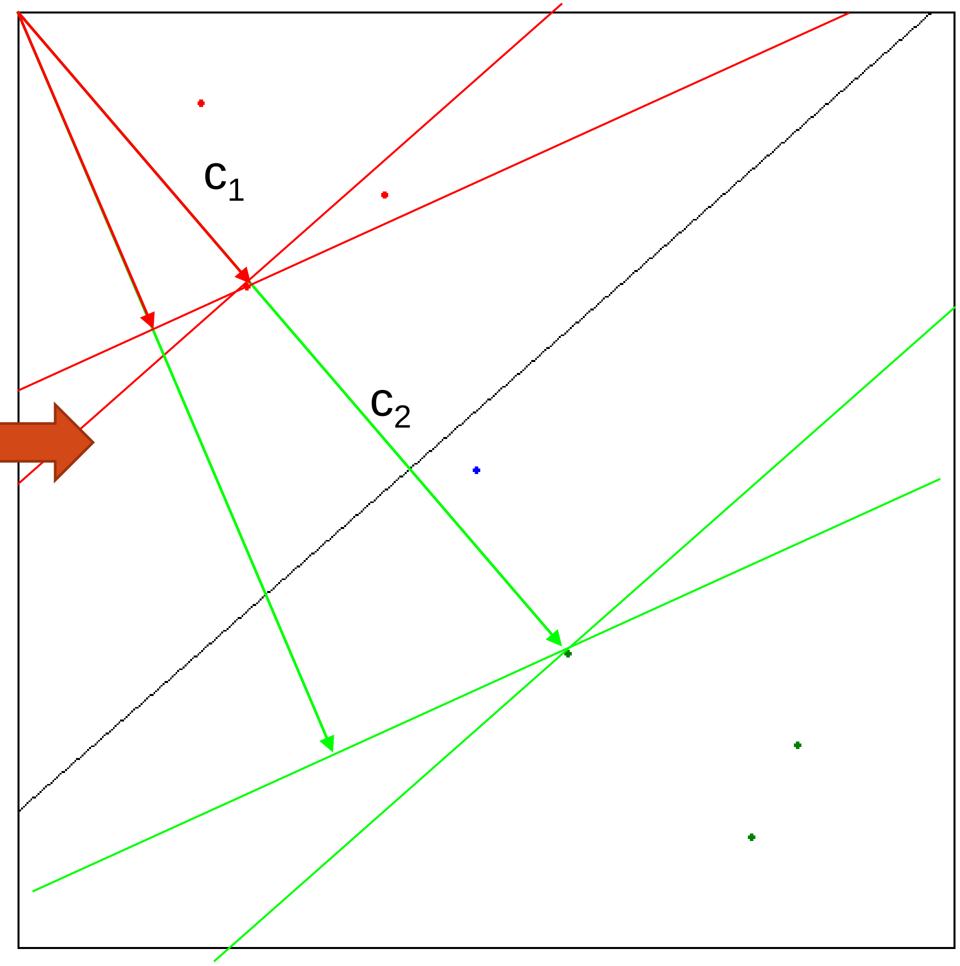
Optimālā atdalošā hiperplakne

- Tagad no visiem tiem nejaušiem vektoriem jāatrod optimālo!
  - Optimālam hipervektoram attālums no vektora līdz tuvākiem klašu objektiem ir maksimāls!
- Kā to tagad realizēt programmā?



Optimālā atdalošā hiperplakne

- **Atceramies, optimālam hipervektoram attālums no vektora līdz tuvākiem klašu objektiem ir maksimāls!**
- Programmā mēs meklēsim divas vērtības – c_1 un c_2.
- Tā kā vektori programmā ģenerējās izmantojot gx un gy – tiem nav pagaidām konkrētu koordināšu telpā. Tāpēc c_1 un c_2 divi ģeometriski izskatīsies tā:



Optimālā atdalošā hiperplakne

- Lai to realizēt programmā, mums jāatrod:
  - c_1 – attālumu līdz tuvākai atdalošai plaknei (tuvāka objekta) A klasē un šī tuvāka objekta kārtas numuru
  - c_2 – attālumu līdz tālākai atdalošai plaknei (tālāka objekta) B klasē un šī tālāka objekta kārtas numuru.

$$c_1 = \min_{A\_class} \sum_{i=1}^{n} x_i g_i$$

$$c_2 = \max_{B\_class} \sum_{i=1}^{n} x_i g_i$$

Optimālā atdalošā hiperplakne

- Programmā c_1 un c_2 un pašus objektus mēs meklēsim tā:

```
c1:=+999999;
c2:=-999999;
 for j:=1 to 3 do
   begin
     tmp:=a[j].x*gx+a[j].y*gy;
     if tmp<c1 then
       begin
         c1:=tmp;
         nr_a:=j;
       end;
     tmp2:=b[j].x*gx+b[j].y*gy;
     if tmp2>c2 then
       begin
         c2:=tmp2;
         nr_b:=j;
       end;
   end;
```

	gx	gy	c1	Nr_A	c2	Nr_B
1	-0.99	0.10	-85	3	-283	1
...	...	...	...	...	...	...
n	-0.13	0.99	176	3	212	2

Optimālā atdalošā hiperplakne

- OK, saģenerējām vektorus, atradām tuvākos/tālākos objektus un attālumus no tiem līdz vektoram. Kas tālāk? Atceramies, **optimālam hipervektoram attālums no vektora līdz tuvākiem klašu objektiem ir maksimāls!** Sanāk, ka optimālo hipervektoru mēs varam atrast, ja papētīsim attālumus c_1 un c_2, jeb Δc.
  $\Delta c = c1 - c2$
- Optimālam hipervektoram Δc ir maksimālais!
- Meklējam maksimālo Δc un atradīsim optimālā vektora kārtas numuru.

Optimālā atdalošā hiperplakne

- Programmā tas izskatīsies tā:

```
dc:=-9999
dc:=c1-c2;
      if dc>dc_opt then
        begin
          dc_opt:=dc;
          i_opt:=i;
        end;
```

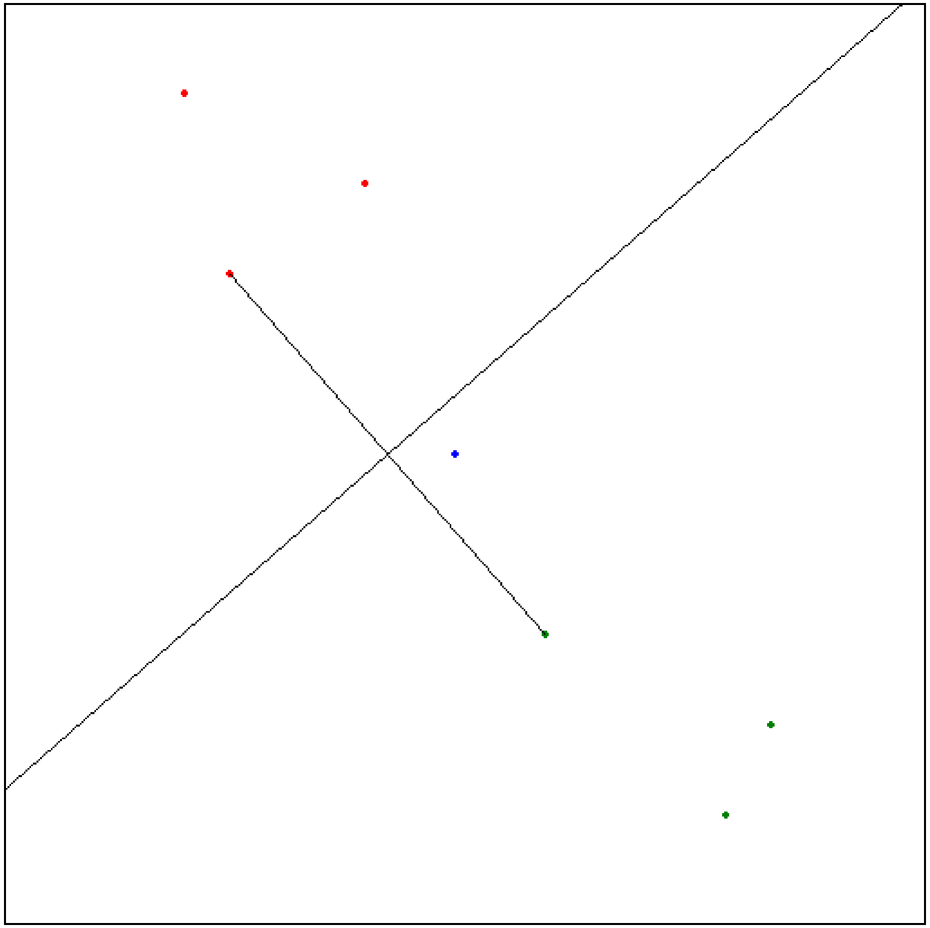
$$\Delta c = (c_1 - c_2)$$

$$\Delta c \xrightarrow[g \in G]{} \max$$

	gx	gy	c1	Nr_A	c2	Nr_B	dc
1	-0.99	0.10	-85	3	-283	1	198
...	...	...	...	...	...	...	
n	-0.13	0.99	176	3	212	2	-36

Optimālā atdalošā hiperplakne

- Savienojam optimālus objektus ar līniju
- Šajā līnijā atradīsim vidējo punktu
- No vidējā punktā izvirzīsim optimālo vektoru – tā ir optimāla atdalošа hiperplakne 2D telpā



Optimālā atdalošā hiperplakne

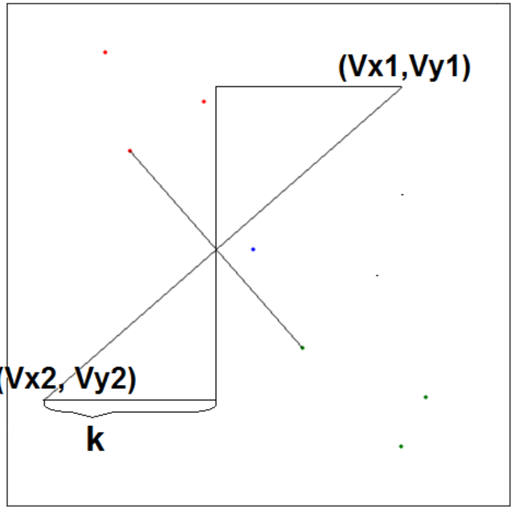
- Programmā optimālās hiperplaknes sākuma un gala punktus var paskaitīt pēc formulām:

Vx1 := k*gy[opt]+(a[opt].x+b[opt].x)/2;

Vy1 :=-k*gx[opt]+(a[opt].y+b[opt].y)/2;

Vx2 := -k*gy[opt]+(a[opt].x+b[opt].x)/2;

Vy2 := k*gx[opt]+(a[opt].y+b[opt].y)/2;



Optimālā atdalošā hiperplakne

- Labi, hiperpplakne mums ir! Tagad, kā noteikt kurai klasei pieder objekts?
- Atkarībā no tā, vai jaunais objekts atrodas pa labi vai pa kreisi no hiperplaknes, tas pieder vienai vai otrai klasei. Programmā to darām tā:

```
Position:= Round((Vx2-Vx1)*(obj.y-Vy1) - (Vy2-Vy1)*(obj.x-Vx1))
if Position<0 then Label1.Caption:= ‘klase A’;
if Position>0 then Label1.Caption:= ‘klase B’;
if Position=0 then Label1.Caption:= ‘A un B’;
```