

# **Universidad Rey Juan Carlos**

**Doble Grado Ingeniería Software y Matemáticas  
Asignatura Geometría Computacional (Cuarto curso)**

## **Práctica 1: Recta y punto**

**Realizado por:** Iván Martín Gómez y Markos Aguirre Elorza

## **ÍNDICE DE CONTENIDOS:**

1. Comentarios previos
2. Problemas
3. Conclusiones
  - 3.1. Opinión personal sobre facilidad o dificultad de la práctica
  - 3.2. Resultados
4. Anexo 1: Código Rstudio

# 1. Comentarios previos

Trabajaremos en el plano ( $\mathbb{R}^2$ ) y una recta en el plano divide el plano en dos semiplanos:

- Opción 1: La recta divide el plano en los semiplanos “izquierdo” y “derecho”
- Opción 2: La recta divide el plano en los semiplanos “encima” y “debajo”

En nuestro caso trabajaremos con la Opción 2 aunque en las clases de teoría trabajamos con la Opción 1. Trabajar con la Opción 2 trae como consecuencia que no consideramos el caso de rectas verticales con ecuación  $x=\text{constante}$ .

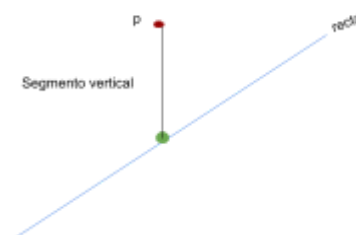
En esta práctica veremos tres métodos para obtener la posición de un punto  $p=(p_1,p_2)$  con respecto a una recta  $y=mx+b$ .

- Método 1: Método Gráfico: este método consiste en graficar el punto y la recta y será el usuario/a la que deberá tomar la decisión de forma visual, de si el punto está encima, debajo o sobre la recta.
- Método 2: Desigualdades: este método consiste en comparar, utilizando desigualdades, la segunda componente del punto  $p$  ( $p_2$ ) con la ordenada de la recta particularizada en la primera componente del punto ( $p_1$ ), obteniendo el siguiente criterio:

*si  $p_2 > mp_1 + b \Rightarrow$  El punto está por encima de la recta*

*si  $p_2 < mp_1 + b \Rightarrow$  El punto está por debajo de la recta*

*si  $p_2 = mp_1 + b \Rightarrow$  El punto está sobre la recta*



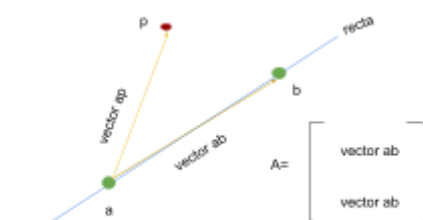
**Nota:** No me parece adecuado llamar a este método el Método de la Proyección ya que en matemáticas es habitual considerar que, el segmento que une a un punto con la proyección de ese mismo punto sobre una recta, forma un ángulo de  $90^\circ$  con la recta y en este caso no es así excepto para el caso particular de estar trabajando con una recta horizontal. Quizás lo que está pasando es que estamos realizando la proyección con un producto escalar diferente al Producto escalar usual. En este caso el nombre del método sí me parece adecuado.

- Método 3: Determinante: este método consiste en utilizar el valor del determinante de la matriz formada por dos vectores adecuadamente escogidos, para obtener el siguiente criterio:

*si  $\det(A) > 0 \Rightarrow$  El punto está por encima de la recta*

*si  $\det(A) < 0 \Rightarrow$  El punto está por debajo de la recta*

*si  $\det(A) = 0 \Rightarrow$  El punto está sobre la recta*



\*Se debe garantizar que la primera coordenada del punto a sea estrictamente menor que la primera coordenada del punto b\*

## 2. Problemas

Obtener mediante cada uno de los tres métodos la posición de un punto  $P=(p_1,p_2)$  con respecto a una recta  $y=mx+b$

Problema 1: Punto:  $p=(1,2)$ , recta:  $y=3x+2$

Problema 2: Punto:  $p=(-1,3)$ , recta:  $y=3x+2$

Problema 3: Punto:  $p=(1,4)$ , recta:  $y=3x+2$

Problema 4: Punto:  $p=(1,1)$ , recta:  $y=5x+1$

Problema 5: Punto:  $p=(2,3)$ , recta:  $y=-2x+1$

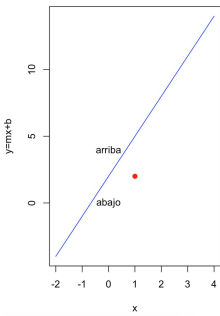
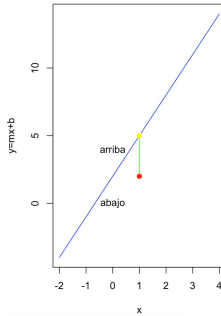
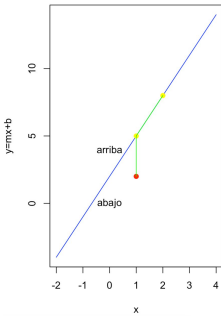
Problema 6: Punto:  $p=(0,-4)$ , recta:  $y=-2x+1$

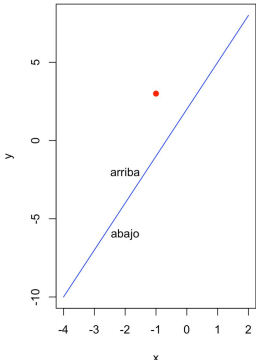
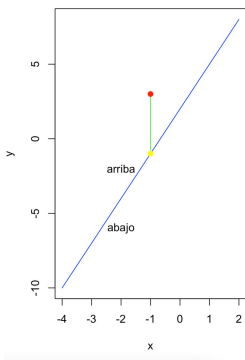
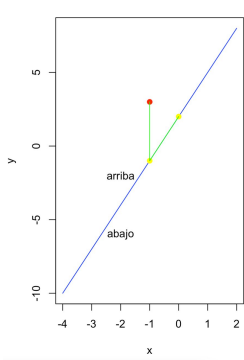
## 3. Conclusiones

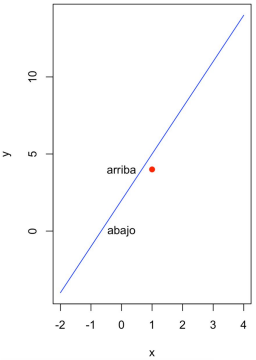
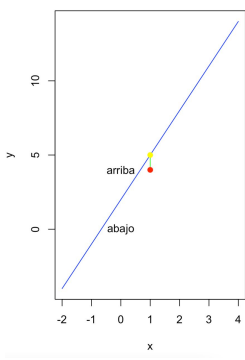
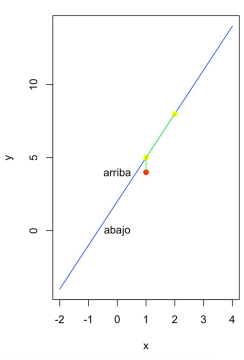
### 3.1. Opinión personal sobre facilidad o dificultad de la práctica

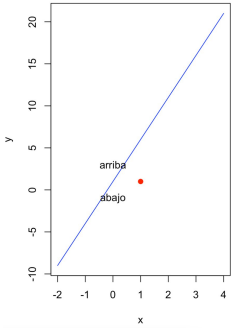
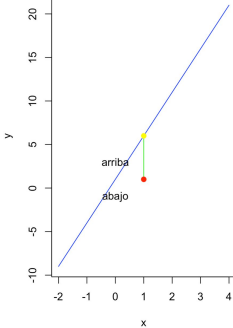
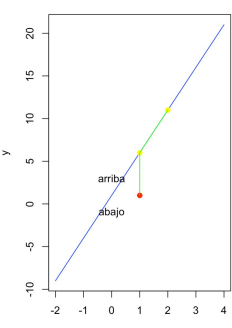
Nivel muy fácil. Al principio cuesta un poco familiarizarse de nuevo con el entorno de RStudio.

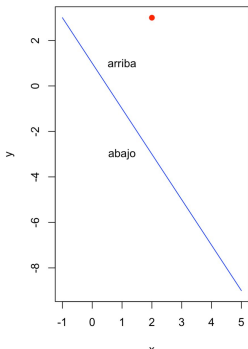
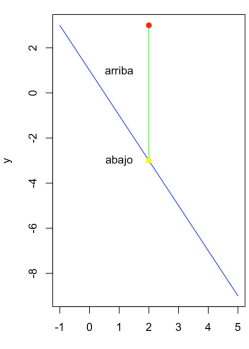
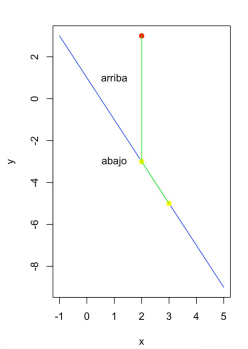
### 3.2. Resultados

<b>Problema 1:</b> Punto: $p=(1,2)$ , recta: $y=3x+2$			
MÉTODO	GRÁFICO	PROYECCIÓN	DETERMINANTE
IMAGEN			
RESULTADO	Usuario decide DEBAJO	Algoritmo decide DEBAJO	Algoritmo decide DEBAJO

<b>Problema 2:</b> Punto: $p=(-1,3)$ , recta: $y=3x+2$			
MÉTODO	GRÁFICO	PROYECCIÓN	DETERMINANTE
IMAGEN			
RESULTADO	<i>Usuario decide ENCIMA</i>	Algoritmo decide <i>ENCIMA</i>	Algoritmo decide <i>ENCIMA</i>

<b>Problema 3:</b> Punto: $p=(1,4)$ , recta: $y=3x+2$			
MÉTODO	GRÁFICO	PROYECCIÓN	DETERMINANTE
IMAGEN			
RESULTADO	<i>Usuario decide DEBAJO</i>	Algoritmo decide <i>DEBAJO</i>	Algoritmo decide <i>DEBAJO</i>

<b>Problema 4:</b> Punto: $p=(1,1)$ , recta: $y=5x+1$			
MÉTODO	GRÁFICO	PROYECCIÓN	DETERMINANTE
IMAGEN			
RESULTADO	<i>Usuario decide DEBAJO</i>	<i>Algoritmo decide DEBAJO</i>	<i>Algoritmo decide DEBAJO</i>

<b>Problema 5:</b> Punto: $p=(2,3)$ , recta: $y=-2x+1$			
MÉTODO	GRÁFICO	PROYECCIÓN	DETERMINANTE
IMAGEN			
RESULTADO	<i>Usuario decide ARRIBA</i>	<i>Algoritmo decide ARRIBA</i>	<i>Algoritmo decide ARRIBA</i>

<b>Problema 6:</b> Punto: $p=(0,-4)$ , recta: $y=-2x+1$			
MÉTODO	GRÁFICO	PROYECCIÓN	DETERMINANTE
IMAGEN			
RESULTADO	<i>Usuario decide DEBAJO</i>	<i>Algoritmo decide DEBAJO</i>	<i>Algoritmo decide DEBAJO</i>

## 4. Anexo 1: Código Rstudio

### Código Método Gráfico:

```
#####
#Title: Posición Punto respecto a Recta utilizando Dibujo y criterio visual Usuario
#Author: Iván Martín Gómez and Markos Aguirre Elorza
#Date: Saturday 13th February 2021
#Descriptions: Dado un punto  $p=(p1,p2)$  y una recta  $y=mx+b$ , decidir si el punto
# está por encima, debajo o sobre la recta. No se contempla el caso de
# de rectas verticales
#####
```

```
funcion_PuntoRecta_Dibujo <- function(coordenada_x_punto =
NULL, coordenada_y_punto = NULL, pendiente_recta = NULL,
ordenada_origen_recta = NULL){
```

```
  p1=coordenada_x_punto
  p2=coordenada_y_punto
```

```

m=pendiente_recta
b=ordenada_origen_recta

#Dibujamos
x1=p1-3
x2=p1+3
x=(x1:x2)
y1=p2
y2=m*p1+b
if(y1<y2){
  y=(y1:y2)
}else{
  y=(y2:y1)
}
plot(x,m*x+b, type='l', xlab="x", ylab="y", col="blue")
points(p1,p2, col = "red", pch=19)
text(p1-1,(m*(p1-1)+b)+2,'arriba')
text(p1-1,(m*(p1-1)+b)-2,'abajo')

}

```

```

#Probamos la función
rm(list=ls())#Limpia Global Environment
dev.off()#Limpia los plots
#Problema 1: p=(1,2), recta == y=3x+2
funcion_PuntoRecta_Dibujo(1,2,3,2)
#Problema 2: p=(-1,3), recta == y=3x+2
funcion_PuntoRecta_Dibujo(-1,3,3,2)
#Problema 3: p=(1,4), recta == y=3x+2
funcion_PuntoRecta_Dibujo(1,4,3,2)
#Problema 4: p=(1,1), recta == y=5x+1
funcion_PuntoRecta_Dibujo(1,1,5,1)
#Problema 5: p=(2,3), recta == y=-2x+1
funcion_PuntoRecta_Dibujo(2,3,-2,1)
#Problema 6: p=(0,-4), recta == y=-2x+1
funcion_PuntoRecta_Dibujo(0,-4,-2,1)

```

### **Código Método Proyección:**

```

#####
#Title: Posición Punto respecto a Recta utilizando proyecciones
#Authors: Iván Martín Gómez and Markos Aguirre Elorza
#Date: Saturday 13th February 2021
#Descriptions: Dado un punto p=(p1,p2) y una recta y=mx+b, decidir si el punto

```



```

#está por encima, debajo o sobre la recta. No se contempla el caso de
#de rectas verticales
#####
funcion_PuntoRecta_Proyeccion <- function(coordenada_x_punto =
NULL, coordenada_y_punto = NULL, pendiente_recta = NULL,
ordenada_origen_recta = NULL){
  p1=coordenada_x_punto
  p2=coordenada_y_punto
  m=pendiente_recta
  b=ordenada_origen_recta

  #Dibujamos
  x1=p1-3
  x2=p1+3
  x=(x1:x2)
  y1=p2
  y2=m*p1+b
  if(y1<y2){
    y=(y1:y2)
  }else{
    y=(y2:y1)
  }
  interseccion=m*p1+b
  puntos_vertical=rep(p1,length(y))
  plot(x,m*x+b, type='l', xlab="x", ylab="y", col="blue")
  lines(puntos_vertical,y, col="green")
  points(p1,p2, col = "red", pch=19)
  points(p1,interseccion, col = "yellow", pch=19)
  text(p1-1,(m*(p1-1)+b)+2,'arriba')
  text(p1-1,(m*(p1-1)+b)-2,'abajo')

  #Salida función por ventana de comandos
  if(p2>interseccion){
    cat("El punto (",p1,",",p2,")"," está por ENCIMA de la recta", "y=",m,"x","+ ",b)
  }else if(p2<interseccion){
    cat("El punto (",p1,",",p2,")"," está por DEBAJO de la recta", "y=",m,"x","+ ",b)
  }else{
    cat("El punto (",p1,",",p2,")"," está por SOBRE de la recta", "y=",m,"x","+ ",b)
  }
}

```

```

#Probamos la función
rm(list=ls())#Limpia Global Environment
dev.off()#Limpia los plots
#Problema 1: p=(1,2), recta == y=3x+2
funcion_PuntoRecta_Proyeccion(1,2,3,2)
#Problema 2: p=(-1,3), recta == y=3x+2
funcion_PuntoRecta_Proyeccion(-1,3,3,2)
#Problema 3: p=(1,4), recta == y=3x+2
funcion_PuntoRecta_Proyeccion(1,4,3,2)
#Problema 4: p=(1,1), recta == y=5x+1
funcion_PuntoRecta_Proyeccion(1,1,5,1)
#Problema 5: p=(2,3), recta == y=-2x+1
funcion_PuntoRecta_Proyeccion(2,3,-2,1)
#Problema 6: p=(0,-4), recta == y=-2x+1
funcion_PuntoRecta_Proyeccion(0,-4,-2,1)

```

### **Código Método Determinante:**

```

#####
#Title: Posición Punto respecto a Recta utilizando Determinante
#Authors: Iván Martín Gómez and Markos Aguirre Elorza
#Date: Saturday 13th February 2021
#Descriptions: Dado un punto p=(p1,p2) y una recta y=mx+b, decidir si el punto
#está por encima, debajo o sobre la recta. No se contempla el caso
#de rectas verticales
#####
funcion_PuntoRecta_Determinante <- function(coordenada_x_punto =
NULL, coordenada_y_punto = NULL, pendiente_recta = NULL,
ordenada_origen_recta = NULL){
  #Control de errores
  if(is.null(ordenada_origen_recta)){
    cat("Error: no se contempla el caso de rectas verticales")
    return -1
  }
  p1=coordenada_x_punto
  p2=coordenada_y_punto
  m=pendiente_recta
  b=ordenada_origen_recta

  #Dibujamos
  x1=p1-3
  x2=p1+3
  x=(x1:x2)

```

```

y1=p2
y2=m*p1+b
if(y1<y2){
  y=(y1:y2)
}else{
  y=(y2:y1)
}
plot(x,m*x+b, type='l', xlab="x", ylab="y", col="blue")
points(p1,p2, col = "red", pch=19)
text(p1-1,(m*(p1-1)+b)+2,'arriba')
text(p1-1,(m*(p1-1)+b)-2,'abajo')

#Construimos punto a=(a1,a2) y dibujamos
a=c(p1,m*p1+b)
points(p1,m*p1+b, col = "yellow", pch=19)
#Construimos punto c=(c1,c2); c1>a1 y dibujamos
c=c(p1+1,m*(p1+1)+b)
points(p1+1,m*(p1+1)+b, col = "yellow", pch=19)
#Construimos vector ac y dibujamos
ac=a-c
lines(c(a[1],c[1]),c(a[2],c[2]), col="green")
#Construimos vector ap y dibujamos
ap=c(a[1]-p1,a[2]-p2)
lines(c(a[1],p1),c(a[2],p2), col="green")
#Montamos la matriz A
A<-matrix(c(ac,ap), nrow=2, ncol=2, byrow = TRUE)
#Calculamos determinante de la matriz A
determinante=det(A)
#Comprobamos signo determinante para decidir si el punto p está arriba, abajo o
#sobre la recta
if(determinante>0){
  cat("El punto (",p1,",",p2,")", " está por ENCIMA de la recta", "y=",m,"x", "+",b)
}else if(determinante<0){
  cat("El punto (",p1,",",p2,")", " está por DEBAJO de la recta", "y=",m,"x", "+",b)
}
}else{
  cat("El punto (",p1,",",p2,")", " está por SOBRE de la recta", "y=",m,"x", "+",b)
}
}
#Probamos la función
rm(list=ls())#Limpia Global Environment
dev.off()#Limpia los plots
#Problema 1: p=(1,2), recta == y=3x+2
funcion_PuntoRecta_Determinante(1,2,3,2)

```

#Problema 2:  $p=(-1,3)$ , recta ==  $y=3x+2$   
funcion\_PuntoRecta\_Determinante(-1,3,3,2)  
#Problema 3:  $p=(1,4)$ , recta ==  $y=3x+2$   
funcion\_PuntoRecta\_Determinante(1,4,3,2)  
#Problema 4:  $p=(1,1)$ , recta ==  $y=5x+1$   
funcion\_PuntoRecta\_Determinante(1,1,5,1)  
#Problema 5:  $p=(2,3)$ , recta ==  $y=-2x+1$   
funcion\_PuntoRecta\_Determinante(2,3,-2,1)  
#Problema 6:  $p=(0,-4)$ , recta ==  $y=-2x+1$   
funcion\_PuntoRecta\_Determinante(0,-4,-2,1)

**FIN PRÁCTICA 1**