

Problemas sobre el Metodo Dual-Simplex

Moyra Fraga-Marquez
Instituto Tecnológico Superior Zacatecas Occidente

INTRODUCCIÓN:

El método dual simplex es otro caso especial de aplicación del método simplex en la programación lineal, en el siguiente trabajo, se mostraran dos problemas utilizando la herramienta AMPL, en donde se va a mostrar de manera mas clara y rápida un resultado favorable.

PROBLEMA:

1) Muchacho que desea vender Limonada y Jugo de Frutas

Un muchacho quiere abrir un puesto de bebidas. Su mama le dice que no puede vender mas de 4 galones de bebidas. El muchacho vende limonada y jugo de frutas, dice que vende la limonada en 2 dolares el galon y el jugo de frutas a 1.50 dolares el galon. La limonada requiere 30 rebanadas de limon por galon y una libra de azucar por galon. El jugo de frutas usa 10 rebanadas y 2 libras de azucar por galon. La mama del muchacho tiene solamente 90 rebanadas de limon y 6 libras de azucar. Enceuntra, ¿Cuantos galones de cada bebida se pueden hacer para hacer la mayor cantidad de dinero o mejor ganancia?.

Solución mediante AMPL

```
AMPL: var limonadas;
AMPL: var jugodefrutas;
AMPL: maximize ganancia:2*limonadas+1.5*jugodefrutas;
AMPL: subject to
AMPL? c1:limonadas+jugodefrutas<=4;
AMPL: c2:30*limonadas+10*jugodefrutas<=90;
AMPL: c3:limonadas+2*jugodefrutas<=6;
AMPL: solve;
MINOS 5.51: optimal solution found.
2 iterations, objective 7.25
AMPL: display ganancia,limonadas,jugodefrutas;
ganancia = 7.25
limonadas = 2.5
jugodefrutas = 1.5
```

PROBLEMA:

2) Distribución de horas de trabajo y de juego de Ping-Pong

Asume que quieres decidir entre formas alternativas de pasar un día de 8 horas, esto es, quieres distribuir tu tiempo. Asume que se te hace 5 veces mas divertido jugar ping-pong que trabajar pero también sientes que debes trabajar por lo menos 3 veces tantas horas como las que jugaste ping-pong. Y ahora el problema es: ¿Cuantas horas debes jugar y cuantas trabajar para maximizar tu función que es la diversión?

Solución mediante AMPL

```
AMPL: var trabajando;
AMPL: var jugando;
AMPL: maximize diversion:trabajando+5*jugando;
AMPL: subject to
AMPL? c1:trabajando+jugando<=8;
AMPL: c2:3*jugando<=trabajando;
AMPL: solve;
MINOS 5.51: optimal solution found.
1 iterations, objective 16
AMPL: display diversion,trabajando,jugando;
diversion = 16
trabajando = 6
jugando = 2
```

CONCLUSIÓN:

Finalmente, gracias a AMPL, el trabajo es mas rapido al momento de desarrollarlo, ya como anterior mente lo vimos esta herramienta es muy especifica al momento de arrojar resultados, siempre y cuando los datos sean aplicados de manera correcta. Es necesario conocer sobre esta aplicacion, ya que se pueden presentar problemas o muy simples o muy complicados, por eso es sumamente importante saber mas sobre esta herramienta para llegar a buenos resultados.